More bargains on Plus 1 & Plus 3 – see centre pages

A Database Publication

electron user

Vol.3 No.6 January 1986 £1

Player's guide to Sphinx Adventure

Acornsoft Database reviewed

6502 graphics – keeping track of sprite collisions

Give your Electron three modes – on the same screen!

2 full length games: Helicopter Rescue & Fruit Worm

A SPECTACULAR ARCADE-STYLE GAME

DeathStar

THE FIRST ZONE.

THE DEATHSTAR BEING BOMBED.

THE WARRIOR ZONE.

CRYSTALS BEING EMITTED FROM A PLANETOID.

DEATHSTAR

is a fast arcade-style space game with stunning graphics and addictive action. Your "intelligent" opponents, the Workers and Warriors, each have several different tasks to perform: mining the Crystals held deep within the Planetoids, constructing the mighty Deathstar from these Crystals, guarding the Deathstar, attacking your ship, and sacrificially intercepting the Starbombs, with some jobs having higher priorities than others. You must amass a large supply of Starbombs by shooting the Planetoids and collecting the Crystals as they splinter off the Planetoids and hurtle through space. Once the Deathstar has been completed, it comes in rapid pursuit and can only be repulsed by swift and accurate use of the Starbombs. If you successfully destroy the Deathstar, you progress on to the Worker Zone, the Warrior Zone, the Planetoid Zone and the Void Zone (in which there are very few Planetoids present), with a bonus screen between each zone. A superb game, destined to become another Superior classic.

SPEECH COMPATIBLE

If you have an Acorn Speech Synthesiser fitted, you will hear spoken phrases: such as "BEWARE, I AM COMPLETE" when the Workers have finished building the Deathstar.

CASSETTE VERSION — £9.95 BBC DISC VERSION — £11.95

Due to phenomenal demand, Deathstar is now available on disc for the BBC micro in a high quality vinyl disc wallet with colour packaging.

WE PAY UP TO 20% ROYALTIES FOR HIGH QUALITY BBC & ELECTRON PROGRAMS

SUPERIOR SOFTWARE LTD.

Dept. RU10, Regent House,
Skinner Lane, Leeds 7
Tel: 0532 459453.

Access

OUR GUARANTEE

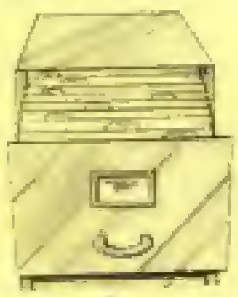
- All mail orders are despatched within 24 hours by first-class post
- Postage and packing is free
- Faulty cassettes and discs will be replaced immediately

CONTENTS

Vol. 3 No. 4 January 1986

electron user

News
All that's new in the ever expanding world of the Electron. 7



Database
For those with information to keep track of we look at databases in general and evaluate Acornsoft's Database in particular. 11

Fruit Worm
Forget the diet! In this superb arcade game you're a hungry worm whose only aim in life is to eat fruit! 15

Tape & Disc
How to avoid problems transferring from tape to programs disc. 19

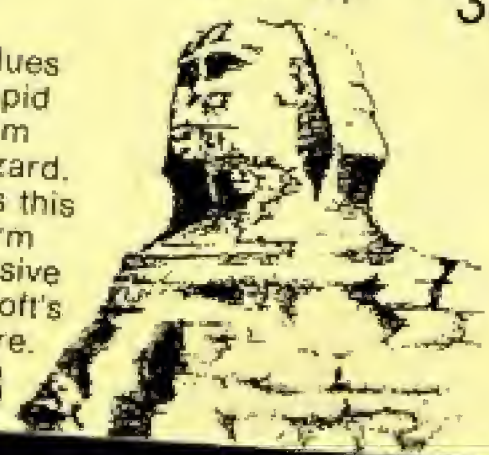
Software Surgery
From Harrier offensives to maths with a story, our reviewers bring you the very latest in software for the Electron. 21

Beginners
We continue our popular series for beginners to programming by taking a detailed look at the last of the Electron's string handling functions INSTR() 25

Machine Code Graphics
Keep your sprites on the screen and watch out for collisions in the penultimate part of our popular series. 30

Merlin's Cave
More tips and clues for all you intrepid adventurers from our resident wizard. There's a bonus this month in the form of a comprehensive guide to Acornsoft's Sphinx Adventure. 33

Helicopter Rescue
Have you the skill and courage to rescue two men stranded on rocks and transfer them to the cliff top? 38



*FX5
Pep up your screen displays with the calls concerned with graphics. 43

Procedures
How procedures are preferable to GOSUBs in structured programs. 48

Order form
Take out an annual subscription, order a back issue, tape, dust cover or binder . . . all on one form. 61

Programming the ULA
In the final part of this series we write to Shiela and get Modes 0, 1 and 2 – on the same screen! 64

Micro Messages
The pages you write yourself. A selection from the many interesting letters you've been sending us. 67

Expand your Electron

Don't miss the biggest money-saving offer we've ever made. You'll find full details in the centre pages.

Managing Editor
Derek Meakin
Features Editor
Pete Bibby
Production Editor
Peter Glover
Art Editor
Heather Sheldrick
Advertisement Manager
John Riding
Advertising Sales
John Snowden
Editor in Chief, Database Publications
Peter Bramald

Published by Database Publications Ltd
Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

Telephone: 061-456 8835 (Editorial) 061-456 8383 (Administration) 061-456 8500 (Advertising)
Subscriptions 061 480 0171 Telecom Gold Mailbox 72:MAG001 Prestel 614568383.
Telex 265871 MONREF G Quoting Ref. 79 MAG001

ABC 26,435 January-June 1985

News trade distribution: Europress Sales and Distribution Limited, 11 Brighton Road, Crawley, West Sussex RH10 6AF. Circulation 0293 27053.

Electron User is an independent publication. Acorn Computers Ltd, manufacturers of the Electron, are not responsible for any of the articles in this issue or for any of the opinions expressed.

Electron User welcomes program listings and articles for publication. Material should be typed or computer-printed, and preferably double-spaced. Program listings should be accompanied by cassette tape or disc. Please enclose a stamped, self-addressed envelope, otherwise the return of material cannot be guaranteed. Contributions accepted for publication will be on an all-rights basis.

Subscription rates for 12 issues, post free:

£12 UK & Eire (Sterling only)
£20 Europe
£35 Overseas

© 1986 Database Publications Ltd. No material may be reproduced in whole or in part without written permission. While every care is taken, the publishers cannot be held legally responsible for any errors in articles or listings.

SOLIDISK EFS COMBINES DISC AND A SOCKET FOR THE WI

Solidisk Double Density DFS is now the ultimate in reliability and supported by the largest amount of software available for the Electron.
Solidisk relies on a good product and a large support network to win the heart of the user.
With over 75 Local Experts, covering England, Scotland and Wales, Solidisk can offer many users regional free fitting and advice.
With an ever increasing catalogue of free software, even users who are new to the Disc system can expect to build up a large library in a fairly short time.
Solidisk Software Support Service already has responsibility for over 50,000 BBC computer users and the ability to give you the best service matched only by the largest companies.
Solidisk Double Density DFS handles both BBC Discs and Electron Discs, in single and double density whereas the Acorn's PLUS 3 can only handle ADFS discs.
Solidisk ADFS has nice features such as automatic disc format sensing, built-in disc formatter and verifier and programmable disc speed.
It also has more than 20 disc utilities built into the ROM.
Standard features for both BBC DFS and ELECTRON ADFS implementations include:
1) Automatic Write Error Correction.
2) Automatic 40/80 track stepping, the ADFS 2.1 will let you read and write 40 trak discs if you have an 80 track drive.
3) Disc repair facilities.
Disc sector editor (*DZAP), memory editor (*MZAP), recover good sectors (*RECOVER) rewrite multiple sectors (*RESTORE), read bad sectors and bad track (*RTRACK), repair and restore bad sectors and track (*WTRACK) and the powerful disc copy (*DCOPY) which is capable of duplicating even some non BBC discs.
4) Tape to disc facilities.
Direct transfer from tapes to disc (*TAPEDISC) will work with all unprotected programs. *TAPELOAD and *TAPESAVE will cope with more difficult ones. Only in some cases (multipart games cassettes) will you need Solidisk tape copier.
5) Wordprocessing facilities.
This facility allows *BOOT and other text tiles to be edited, saved and printed in any screen mode.
6) Automatic disc format sensing.
On Shift-Break, the STL ADFS 2.1 will detect the disc format and use the right BBC DFS or Electron ADFS to run.
On the Electron ADFS side, the 2.1 ROM also has some very nice features:
1) Extensive Disc formatting facilities.
*FORM40, *FORM80, *FORM160 and *WFORM (for the Winchester) are available to handle any disc drive.
2) Disc verifying facilities.
*VERIFIFY will check all disc sizes including Winchester for media defects.
3) Number of opened channels.
This is the star feature of Solidisk ADFS.
This facility (*OPEN) allows you to specify how many files will be opened in a program, thus maximising the available RAM while avoiding buffer page swapping as on the Acorn ADFS.
It leaves PAGE at &1900 for most programs, gives more room to View and Viewsheet and avoids unnecessary conversion work for many programs originated for the BBC DFS to be run on your Electron.

On the BBC DFS side, the STL ADFS 2.1 handles both single and double density and in addition, it supports:
1) Unlimited catalogue entries.
2) Unlimited filesize.

THE SOLIDISK 16k SIDEWAYS RAM:

Solidisk Sideways RAM is an almost indispensible add-on for the Electron with disc drives.
The Sideways RAM occupies the same memory area as the BASIC or ADFS ROM in the micro's memory map. This means that Sideways RAM can run almost any ROM type software, including languages, utilities and games.
Sideways RAM is notably invaluable to run games and specially "MEGAGAMES".
Games and programs run at 2MHz clock speed in Sideways RAM, if loaded into the Electron RAM, they can only run at 1MHz clock speed, ie half the speed of Sideways based games.
Megagames are too large to be run on the unexpanded Electron. They use extensively 8 colour high resolution screen (mode 2), background music, sound and high speed sprites.
Solidisk supply free software to maximise the use of Sideways RAM on the Electron. These include Wordprocessor, Spreadsheet, Database, Toolkit, Machine Code Monitor, Printer Buffer, Sprites, Playtunes, Virtual Memory Processor, VDU Replay, Screen Effects, digitised pictures etc . . .

THE WINCHESTER SOCKET:

Solidisk has the most powerful Winchester system for the BBC computers and the Electron. The Winchester system can provide from 20 Megabytes to a theoretically possible 1300 Gigabytes of storage, directly on line with the Electron.
The same Winchester unit can be used on the BBC B, the BBC PLUS and the Electron without any change.
You can read more about it in BBC Micro User or in Acorn User Magazines. Price of a 20 Megabytes system is only £700.00 + VAT (£805.00).

SOLIDISK TECHNOLOGY LIMITED, 17 SWEYNE AVE, SOUTHEND-ON-SEA, ESSEX SS2 6JQ. TEL. SOUTHEND (0702) 354674 (10 lines).

UPGRADE, 16K SIDEWAYS RAM
NCHESTER FOR ONLY £59.00

SOLIDISK SPECIAL MITSUBISHI DISC OFFER:

This offer comprises:
— One 80 track Double Sided (640 kbytes) 3.5" Mitsubishi disc drive with its own PSU. Cased in beige.
— Solidisk EFS Disc Upgrade, 16K Sideways RAM and Winchester socket.
— One software package containing four 3.5" discs, detailed below.
— Full one year guarantee and 2 manuals.

PRICE: £200.00

You can also order as many Megagame Packs at the same time as you like. Each Megagame Pack consists of three 3.5" discs and contains on average 20 games.

THE SOFTWARE:

The software contains everything to start a library: the big four (Database, Wordprocessor, Spreadsheet and Graphic), Utilities and Games.

Database:

Solidisk Database is very easy to understand and use. You are presented with a 15 option Menu. Each option will lead to a new Menu and so on. Mode 3, 80 column screen is used throughout so that what you see is what will be printed on paper. With Solidisk Database, you can create as many records as you like, each record can be up to 15 fields of up to 60 characters. You can sort, search, index, mailmerge, append, create subset, calculate etc. Solidisk use the same Database to process all your orders.

Wordprocessor:

Solidisk Wordprocessor is WYSIWYG type (What You See Is What You Get) and has all the commands of a professional tool. It features 80 column screen, on screen justification, page numbering, search and replace, word count, free space, Wordstar like editing commands: insert and overwrite, block mark, move, copy, delete, save, load to cursor, * commands etc..

Spreadsheet:

Solidisk Spreadsheet is also Menu driven and has the same file structure as Solidisk Database. You can have as many rows and columns as you like, each column can be as small as two characters wide or as big as 70 characters. All maths functions are

supported. Recalculate, Replicate, Print, Print If, Sort, Search, Define Zone, Mailmerge, Text Input/Output etc . . . are included.

Toolkit:

Solidisk Toolkit is almost indispensible for Electron programmers, it has 24 star commands (Status, Rwipe, LVAR, Move, Search and Replace, Expand, Salvage, Keyload etc . . .).

ORDER FORM

PRICE LIST and ORDER FORM.	Price	P&P
Diskettes.		
Verbatim 3.5" box of 10	£40.00	£1.00
Software Pack	£10.00	£1.00
Solidisk EFS	£59.00	£1.00
Disc Offers.		
640k Mitsubishi MF453 Disc Outfit	£200.00	£3.00
1.3 MB Fantastic Offer	£310.00	£3.00
20 MB Winchester system	£805.00	£10.00
	Total =	

Name: _____

Address: _____

_____

_____

Payment ☐ Cheque ☐ Cash or Postal Orders

☐ Bank transfer ☐ Other means.

Access or Barclay Credit Card Account: ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

You can also order by phone 0702-354 674 9AM-5.30PM

SEND TO:
SOLIDISK TECHNOLOGY LIMITED,
17 SWEYNE AVE,
SOUTHEND ON SEA,
ESSEX SS2 6JQ.

We reserve the right to change specifications and prices for improvements.

MIRRORSOFT

STRIKE FORCE HARRIER

Designed in co-operation with British Aerospace, Strike Force Harrier is a full colour flight and battle simulation complete with manual, flight pattern diagrams, and free poster.

Features:

Multiple ground targets	•	Multiple air targets
Ground detail	•	3-D mountains
Sky detail	•	Moving horizon
Weapon choice	•	V/STOL
Multiple missions	•	Strategic planning
Practice mode	•	Realistic air combat
Full aerobatics	•	HUD

Can you meet the ultimate challenge?

Publication October 15th

On cassette for BBC B and Electron £9.95

On disk for BBC B £12.95

MIRRORSOFT

Maxwell House, 74 Worship Street, London EC2A 2EN. Tel: 01-377 4644 (24 hours)

Trade orders: Mirrorsoft Ltd, Purnell Book Centre, Paulton, Bristol BS18 5LQ.

electron user NEWS

Show sales will hit £1 million mark

John Huddlestone . . . confidence in the Electron's future

A POLL of exhibitors conducted at the close of the November Electron & BBC Micro User Show revealed that revenue from the event is expected to top £1 million.

Some 90 per cent of the companies contacted agreed to disclose their figures with the strict proviso they were for statistical purposes only.

"We have never experienced the like of it before", one leading exhibitor told *Electron User*. "It was a pure business bonanza".

Company after company reported record takings as once again more than 20,000 visitors went through the turnstiles during the four day event.

ELECTRON WAS THE BIG DRAW

On several occasions during the Saturday and Sunday openings admission had to be temporarily suspended as the "house full" sign went up at the New Horticultural Hall, London.

But even on the relatively quieter opening day – the Thursday – stallholders were swamped with customers.

One company – Advanced Memory Systems of Warrington – took more money during the first day alone than for the whole of any previous show they had attended.

"It was just incredible", said Nick Pearson of AMS, "and it carried on that way for the rest of the show".

The show's organisers, Database Publications, believe much of the reason for the record sales stems from the hundreds of special offers available.

Bargains

"People are definitely getting more and more price conscious these days, so they were particularly keen to snap up all the bargains", said Derek Meakin, head of Database.

Most exhibitors contacted at the end of the show expressed surprise about the flood of interest in Electron products.

"For the first time it seemed to be more Electron-orientated than BBC", said Andrew Hildig, of Slogger Software.

"The Electron is now coming out of the shadow of the BBC Micro".

This view was supported by Tom Crossley of Kansas City Systems: "It seems this is now the day of the Electron", he told *Electron User*.

MASS LAUNCH FOR ELECTRON

A PACKAGE of nine new products for the Electron – the largest single mass launch ever for the machine – comes from Advanced Computer Products.

"We are once more demonstrating our confidence in the future of the Electron", said John Huddlestone of ACP at the November Electron & BBC Micro User Show launch.

"Judging by the response we had the users feel exactly the same way. For Electron users seemed to outnumber even BBC Micro users at this year's show".

The new product range includes three that are now on sale and a further six that are, as yet, in prototype form.

Currently on sale are:

- Advanced Word Power, a word processor claimed to combine ease of use for beginners and sophisticated facilities for serious users. It is selling for £40.
- A range of second drives for the Electron with power supply units from £140.
- A second drive adapter for the Plus 3 for £7.95.

Still to go into production:

- Advanced Eprom Programmer written for the BBC Micro but now adapted for the Electron.
- Electron disc interface featuring 1770 DFS as standard running with page at &E00.

Turn to Page 8

'Over the top' Whoopsy heads for the Top Ten

A NEW game for the Electron has got the thumbs down from the major distributors on the grounds that it is too rude.

Yet Whoopsy, from Shards Software, is on the verge of becoming a bestseller anyway based on mail order sales only.

Now Steve Maltz, of Shards, is claiming that this is simply a case of the distributors lacking a sense of humour.

"We have been amazed by their reaction", he told *Electron User*. "After all, the program was written by a 54-year-old grandfather and retired businessman.

"Naturally it won't be to everybody's taste, but to virtually place an embargo on it is all a bit silly".

Problem

The problem stems from the fact that Whoopsy – as the name implies – involves a baby (the player) attempting to frustrate its mother's attempts at potty training.

However it is apparently not the graphics which have caused offence but rather the accompanying sound effects.

"To have what are obviously lavatory noises in it we feel goes rather over the top", said one distributor.

Meanwhite Whoopsy was reported to have sold in large numbers at the Electron & BBC Micro User Show.

"Judging by its success to date, it would seem to me that the distributors themselves have made a bit of a Whoopsy by not handling the game", said Steve Maltz.

Steve Maltz... "amazed by the distributors' reaction"

TELETEXT ADAPTER

ELECTRON users can now receive teletext and run Mode 7 BBC Micro games thanks to Morley Electronics.

The Tyne and Wear company has produced an Electron version of its teletext adapter, which simulates a Mode 7 screen.

It has an inbuilt red/green/blue output and an optional VHF modulator for those without a monitor.

Price: £149.95.

'Electron close to BBC'

From Page 7

so there is no loss of memory.

- User port which provides the same I/O as the BBC Micro. A 1MHz bus provides additional I/O.
- Tube interface for the Second Processor which provides 64k memory and dramatic speed increase.
- RS232 interface to enable the Electron to go on-line.

"Interest shown in all these products at the show was simply phenomenal", John Huddlestone told *Electron User*.

"The reason for this is that people have suddenly realised that the Electron is getting as close as it can to the BBC Micro".

WHIZZKIDS TAKE SHOW CLASS

COMPUTER whizzkids from throughout the Greater London area – some as young as three – were recruited to act as "teachers" recently. And it had nothing to do with the current industrial action.

Joanna Green, 8, and Mark Freedman, 7 – both pupils at Moor Lane Junior School, Chessington – were among the Greater London youngsters recruited as "teachers" for the November Electron & BBC Micro User Show.

In fact they only held their posts during the November Electron & BBC Micro User Show in an "electronic classroom" run by Symbiotic Computer Systems.

The children had been selected by the show's organisers, Database Publications, to explain the mysteries of the micro to non-computer buffs among the adult visitors.

In all some 20 schools were represented.

But what each of the children had in common was that he or she was considered to have the makings of a child prodigy on either the Electron or the BBC Micro.

All the children involved were suitably impressed with their intended show uniforms of mortarboards and gowns. However the canes they were presented with left them totally puzzled. They had never seen one before.

Disc games arrive

GAMES on disc for Electron owners with Plus 3 are being offered by Blue Ribbon Software.

The programs are Astro Plumber, Diamond Mine, Diamond Mine II, Castle Assault and Nightmare Maze which usually retail at £2.50 each. Disc price for all five is £9.95.

Packing in adventure

TEXT adventure Rick Hanson, from Robico Software, is now available for the Electron.

The game is identical to the BBC Micro version except that it runs in Mode 4 without coloured text and uses screen memory to store some of the program.

Text compression techniques allow nearly 30k of descriptions to be packed into the Electron along with 220 locations and an advanced sentence interpreter. It costs £9.95.

Budgie's revenge

DESCRIBED by one enthusiastic reviewer as "an exciting mixture of Space Invaders and Defender", alien bashing Video's Revenge from budget software house Budgie has been converted for the Electron. Price: £2.99.

AS FIRST

THE first Electron game to be marketed by Audiogenic Software under its agreement with Icon Software is Caveman Capers.

The multi-screen action program now comes back-to-back on the same cassette as the BBC Micro version and costs £7.95.

Electron hits No 2 spot in the micro sales chart

DESPITE the prophets of doom and gloom, the Electron has zoomed to number two in the latest industry survey of home computer sales with 20 per cent of the market.

Written off months ago by the computer press – with the exception of *Electron User* – the resilient machine is thriving in the cut-throat atmosphere of the High Street.

And with major retail outlets Dixons and Currys offering 100,000 Electrons bundled with cassette recorder and software for under £100, the run-up to Christmas could result in an even bigger slice of the market when the final holiday sales figures are tallied.

The continuing success of the Electron is not hard to figure out if you can avoid the tunnel vision of the pundits – blinkered by Acorn's recent financial problems – whose reports of the machine's death have been exaggerated repeatedly.

The Electron has had the support of some very powerful allies – including *Electron User*, major retailers, a stubborn hard core of software and peripheral developers, Acorn itself, and – not least of all – its enthusiastic user base.

As long ago as October *Electron User* reported that its circulation had risen by 46 per cent in six months to more than 26,000 copies an issue – a clear indication that interest in and support for the Electron was growing rather than declining.

In the same issue Acornsoft announced it would soon resume development and production of new software for the Electron. The previous month the Acorn subsidiary had pledged that all its future programs would run on the Electron as well as BBC Micros.

At about the same time the *Electron User* campaign to get more educational software on to the shelves of computer retailers was boosted by the formation of British Educational Software Associates, an alliance of eight major publishers.

Assurance

And in November Acorn's new managing director, Brian Long, gave *Electron User* the assurance: "We are not ceasing production of the Electron" – a promise since repeated by other Acorn executives.

Also in that issue, Dixon's financial director Egon von Greayez stated: "We expect to sell a lot of Electrons in the run-up to Christmas – it is the right machine now at the right price".

Belief in a future for the Electron was also dramatically demonstrated by greatly increased attendances at Database Publications' 1985 Electron & BBC Micro User shows in London and Manchester.

Perhaps the most telling developments are those that have brought the Electron into line with more powerful micros and opened up a whole new world of serious applications to its owners.

The Plus 1, Plus 3, sideways ROMs – and latterly the long-waited communications package developed jointly by Acorn and Pace Micro Technology – have combined to give it a degree of sophistication undreamed of a year ago.

All of which explains why the Electron is entering a new year second only to the Spectrum in current total sales and with its future seemingly assured.

When *Electron User* told Acorn's marketing boss John Caswell of the Electron's standing in the marketplace he said: "It's tremendous news – we are obviously delighted.

"For so long the general computer press has been putting out only the worst news about Acorn.

"But things were never as bad as they were being painted and we always had faith that there was a lot of life left in the Electron.

"Now this industry survey has proved we were right".

Bob Hillyer (left) and Mike Male, co-authors of steam railway simulation Southern Belle from Hewson Consultants, take to the footplate of Great Western tank locomotive 1466 to announce that the program is now available for the Electron at £7.95.

ELECTRON

DOUBLE DENSITY DISK INTERFACE

SUPER SAVER

You've seen the Rave Reviews, now Cumana
offer their highly acclaimed
Double Density Disc Interface for the Acorn Electron
at the staggeringly low price of:

£74.95

INCLUSIVE OF VAT & DELIVERY!

Just plug the Cumana Interface into the Plus 1 expansion unit on your Electron and hook-up a 3½ inch or 5¼ inch Cumana disk drive

It's as simple as that!

ADVANTAGES OF THE CUMANA INTERFACE:

- Enables use of 3.5" or 5.25" disk drives – Single or Dual (or mixed)
- Single density converter enables existing non-protected Acorn 5.25" software to be read and converted to Cumana format (provided the software is compatible with the Electron's operating system, it will then run)
- Facility to date stamp files
- Does not use Electron RAM
- Allows up to 90 file names per disc
- Copes with 40/80 track as well as Single or Double sided drives
- Allows the use of Wildcards in file name descriptors
- Allows up to 8 simultaneously opened files
- Provides additional sideways ROM socket

Note: Needs Acorn Plus 1 Expansion Unit

ORDER NOW TO AVOID DISAPPOINTMENT!

* Cumana and Acorn are registered trademarks.

CUMANA
The best name in memory

Cumana Limited
The Pines Trading Estate, Broad Street
Guildford, Surrey GU3 3BH
Tel: (0483) 503121
Telex: 859380 Fax No. 503326

ORDER FORM (Cumana Interface for Electron – Electron User January)

Please rush me ..... (qty) disk interface(s) for the Acorn Electron at £74.95 each

I enclose a Cheque/PO for £..... or debit my Access/Visa* Card – Account No: .....

Name: .....

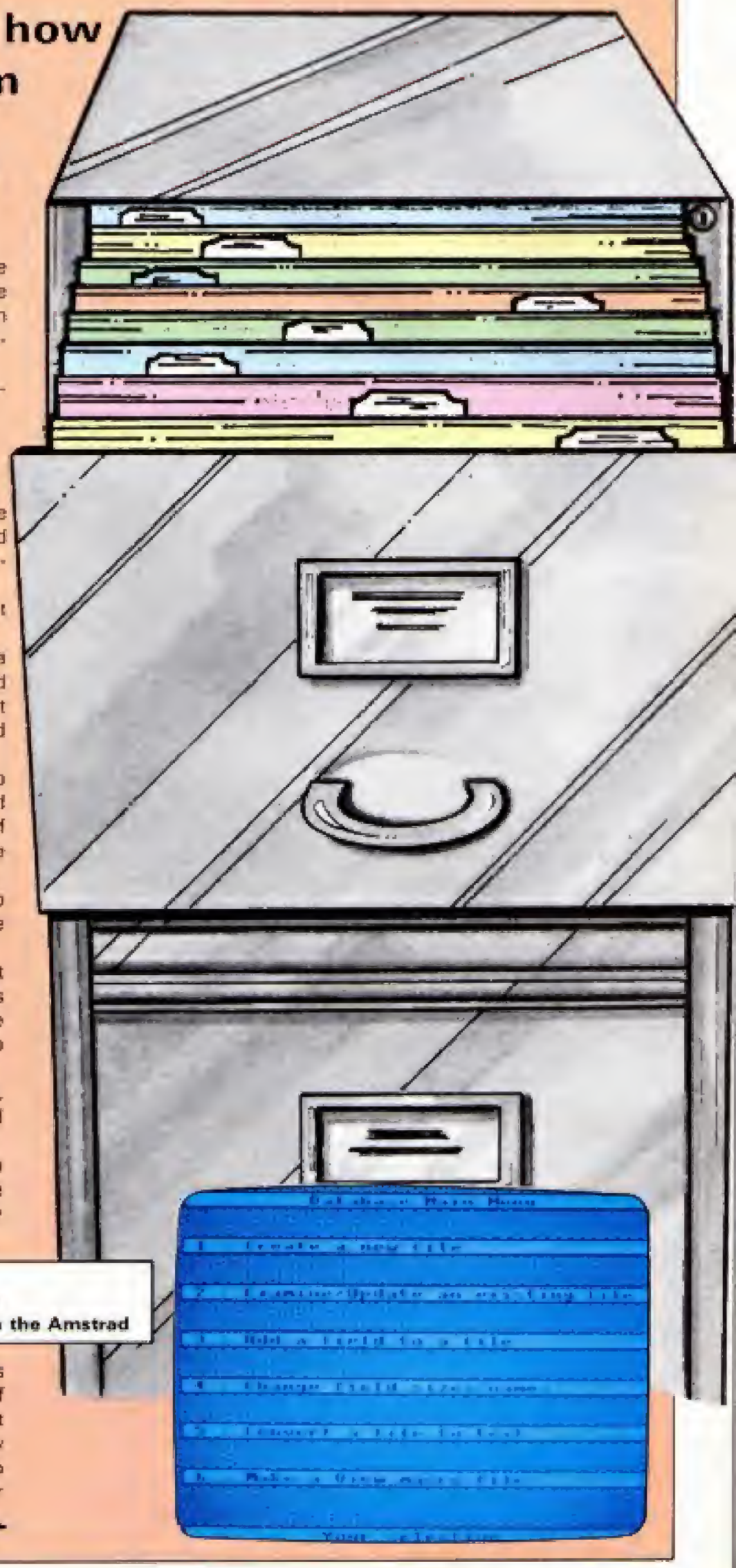
Address: .....

County: ..... Postcode: ..... Telephone No. .....

*Delete as appropriate

Database:

What a database does, and how the Electron Database from Acornsoft measures up. Reviewed by PETE BIBBY



THE excellent Penguin Dictionary of Microprocessors defines a database as:

- A file of data structured to allow a number of applications to access the data and update it.
- Any file which might sound more important as a database.

The question I asked myself when I received the Acorn Electron Database was, which of the two was it?

Having come across some dire databases in my time, I approached the package warily.

My needs are simple – all I want in a database is something that will allow me to record information about things, say my birdwatching, retrieve and update it as necessary, and also use the information in a variety of ways.

Rarely have I found a database that comes up to scratch, so I'm pretty difficult to please.

With the Acorn Database, what you get for your money is a 3.5 inch disc for use with a Plus 3 and a manual.

The disc gives you all the programs needed to create and maintain a database, while the manual tells you how to use them.

To start using the Database, all you have to do is to put the disc in the drive, boot it and a menu appears on the screen giving you a set of six choices as to what to do next.

Incidentally, booting the disc is not as violent as it might seem. All it means is that the disc has been prepared with a previous:

```
*OPT 4 3
```

so that when you hold down Shift, then press and release Break and, finally, release Shift, one of the programs on the disc is CHAINed automatically.

As you'll see if you catalogue the disc with:

```
*MOUNT
*.
```

the database consists of some 20 programs. These are used to create, maintain and analyse files of information.

So what is a file and what information can it hold?

At its simplest a file is just a collection of data, structured in a way that allows you to get at the information easily and flexibly.

A particular file is made up of a series of records, and these records are made up of all the items you want to store in the database.

This is the data you want to record, hence the name record.

Each record has a strict format, or "shape", and it is the database's job to help the user establish this format to meet his or her needs.

If that all seems a bit formal, the following examples should make it clearer.

In the simplest case each record would consist of one item. A file of magazines may consist of the records:

Record 1	**Electron User**
Record 2	**The Micro User**
Record 3	**Computing with the Amstrad**

Here each of the records has only one field, the name of the magazine. However that information could be as easily stored in an array. There's no real need to use a database for

→

From Page 11

such a simple case.

Usually a record is more complicated, and contains two or more items of information linked together. Examples that spring to mind are records of people's names along with their ages or a stock list where each record gives the name of an item, its price, and the number left in stock.

These separate categories of information that go into making up a record are called the fields of a record.

In the case of the name and age example, the records are made up of two fields, one containing the name and the other the age as shown in:

	Field 1	Field2
Record 1	Pete	35
Record 2	Bodger	4
Record 3	Spot	3

So to recap, a file is made up of a set of records and each record is made up of one or more fields. The fields in each record contain the data we wish to record.

Notice that the stucture of each record is the same. They all have the same number of fields in the same order.

What makes them different, of course, is that the information contained in each field varies from record to record.

The Electron Database is a set of programs that allows you to create such files and then amend and analyse them as needed. Again an example should make things clearer.

In my case, being an avid avian enthusiast, I want to keep a record of all the birds I've seen this year. There will be one record for each of the birds spotted.

Each record will consist of the species' name, where I've seen them, the month and any comment I want to make about a particular species. In other words, there'll be four fields in each record, as shown in Figure I.

Each field is identified by a name which can be up to seven letters long, while the entry in each field can range from 2 to 240 characters.

When I used Database to set up my file I named my first field "Species" and set its maximum length to 20 characters. The next field I decided would be called "Place" and be up to 10 characters in length, and so on for the remaining two fields.

Record	Species	Place	Month	Comment
1	REDTHROAT	SPURN	MAY	TICK
2	FLAMINGO	TITCHWELL	OCTOBER	ESCAPE?
3	BITTERN	CLEY	OCTOBER	TICK
4	BLACK TERN	BLACKTOFT	APRIL	FOUR OF THEM
5	RED-NECKED GREBE	CLEY	OCTOBER	TINY. TAME.

Figure I: The fields and data for my birding database

If this sounds complicated, it isn't. The program takes you through creating a file step by step, asking you how many fields you want and then requesting information about each field in turn.

It's surprisingly easy, though it does help if you've decided how you want your records structured before you run the program.

Once all the details of the record are established you're asked for the name of the whole file, and the micro creates that file on the disc. You can then add records, delete them or alter the information in the fields as required.

The individual records making up the file are displayed one at a time in the form of cards. Each "card" holds one record, the record consisting of the field names and the data you've put into those fields. Figure II shows this diagramatically.

As there can be up to 32 fields, each of which can be up to 240 characters in length, you can see that some records could overflow the card. What happens then is that the record is spread over two or three cards.

Taking the case of my ornithological database, each record would fit on one card. In the whole file there would be five cards, one for each species. On each card there are the four fields showing the species, where it was seen, when and any comments.

Once you've got a file of records like this, you can do all sorts with it. For a start, you can change the information held in each record, altering whatever is written in the fields by using the cursor control keys to target the area to be changed and then typing as usual.

Once you've updated the file it can be saved again. Also new records can be added or old ones deleted easily and speedily, so your database can grow and alter with time.

However there's a lot more to it than that. So far we've only seen the records being used as a list. We enter the data and it stays there until it's deleted or amended. In many ways, paper, a pencil and an eraser could do the job as well.

Where a database scores is in what you can do with the information once you've got it in a file.

For a start you can sort it into alphabetical order at the press of a key. So I could rearrange my bird database so that the card with the Bittern data comes first, the Black Tern next and so on.

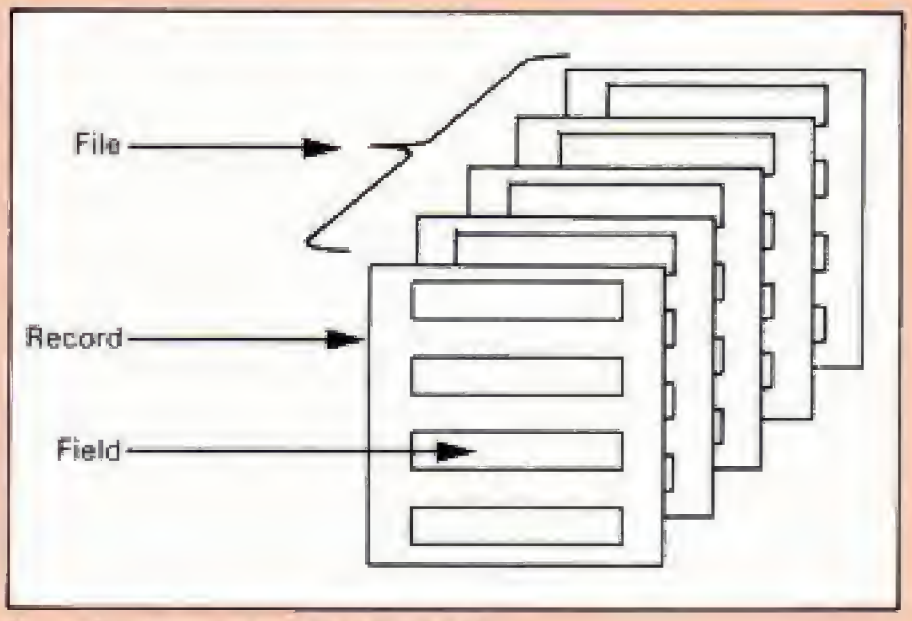


Figure II: Simulating the "card" system

However the sort facility isn't just available on the first field, you can choose any field to sort on. So I could have the records in the alphabetical order of the months, or the order of the places, the card with Blacktoft coming first, the one with Titchwell last.

You can even sort on different fields, giving one field precedence and, if two data items were the same (as in Cley) using whatever is in the other field as a "tie-breaker" to choose what comes next.

In fact you can sort on all the 32 allowable fields, giving each a different priority. I doubt if anyone would take advantage of that too often!

If you've ever tried any sorting, you'll probably have come across the weird case where a sort routine tells you that 123 comes before 5. This is because only the first characters in each have been compared and, of course, 1 comes before 5.

The Electron Database allows you to sort numbers properly. You simply specify which fields are to be treated as numbers with a letter N. Unless you've fallen into the 123 before 5 trap, you won't realise what a gift this is.

As well as sorting the file, you can search it for particular items.

If I wanted to find out where and when I'd seen a Redthroat all I have to do is press a key, specifying that I want all the records with REDTHROAT in the first field.

The Database then looks through the whole file and

collects together all the relevant records. In this case there's only one.

As with sorting, the search facility is much more powerful than it seems at first. By creating patterns made up of characters and symbols with special meanings, you can hunt through the file for all the records that fit that pattern.

For example, if I wanted all the records of birds I'd seen whose name begins with RED I could tell the Database to select them from the file with the template pattern RED*. The asterisk is a special character that means "and any other characters in this field".

What you get from this is what is known as a subset of the original file, in this case containing two cards with the information:

REDTHROAT	SPURN	MAY	TICK
RED-NECKED GREBE	CLEY	OCTOBER	TINY. TAME.

This can be saved as a data file if needed and all the other Database operations can be performed on it.

The asterisk is just one example of a special character that can be used to create a search pattern. There are several of these characters which, taken together, give the Database an amazingly powerful and flexible method of searching files for particular sets of records.

And like the sort, the search isn't just confined to the first field in a record, it can work on any field or any combination of fields. All you have to do is to supply a pattern for each field.

If I wanted all the birds I'd seen at Cley I'd just use the pattern CLEY in the Place field and, at the press of a key, the Database would find all the records that matched this. I'd end up with a subset of the original file made up of the following records:

BITTERN	CLEY	OCTOBER	TICK
RED-NECKED GREBE	CLEY	OCTOBER	TINY. TAME.

As you'll have seen from the descriptions of the search and sort facilities the Database has, the structure of the records has a powerful effect on what it can and cannot do.

This can cause problems. For example, in my birding file I couldn't search out the birds I'd seen on rainy days with an East wind blowing because I have no weather field.

Similarly I've only got 10 characters in my Place field. What if I'd been birdwatching at Oswaldtwistle?

As you can see, the fixed record format that allows the Database to search and sort records could be a bit of a straitjacket.

The ideal would be to have all the fields you need, all of the right length at the start. However no one's perfect, and the Database realises this. It allows fields to be added to records or existing field names and lengths to be altered quickly and simply.

And once you've got all the records you want containing the information you want, you can print them out, choosing which fields of the records to print out and also the format of the hard copy.

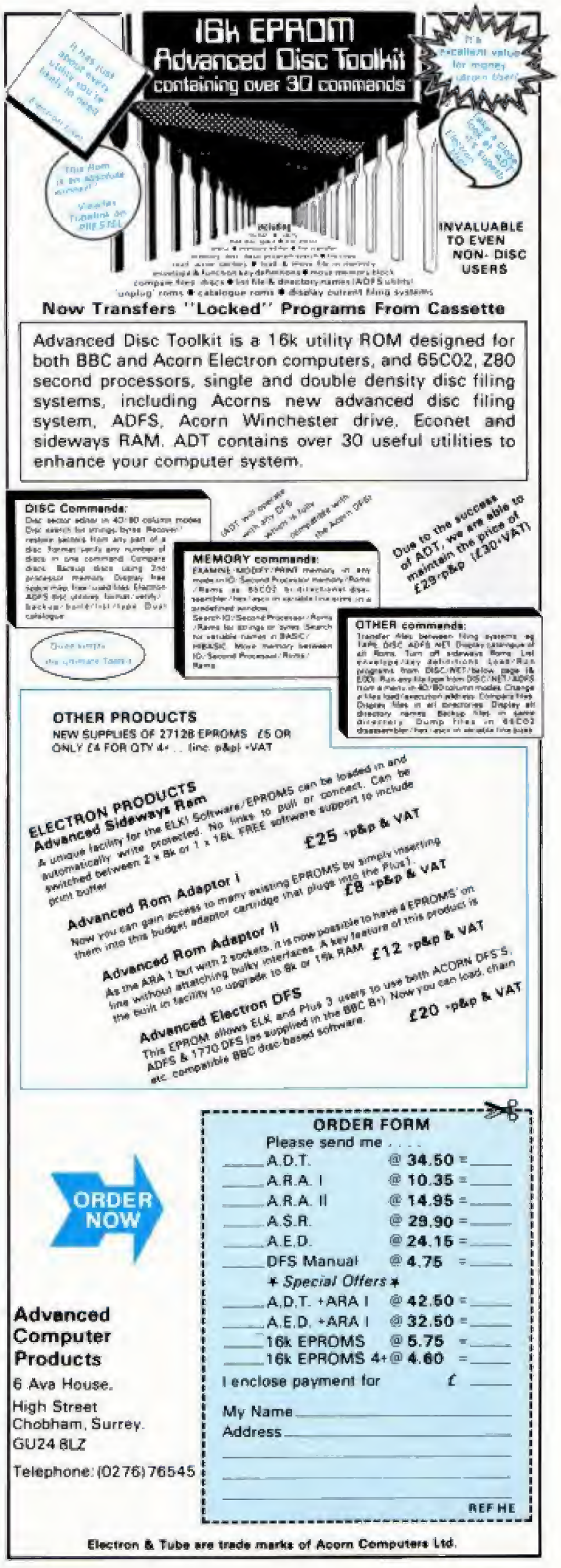
I think that the Electron Database would be great value for money if these were all the facilities it has, but in addition, there are three more.

The first allows you to perform statistical analyses of numeric data, producing the sum, mean and standard deviation of a set of figures.

The second is that the Database has the ability to create a text file copy of one of its data files. This text file can then be loaded into a word processor and treated in the way you would any other piece of word processor text. Hence datafiles can be used in documents.

Still with word processing, the Database can produce a file to be used as parameters for a VIEW macro. Briefly this means that the VIEW word processor holds a piece of text with gaps in it and reads the parameter file created from the data file to fill in these gaps.

A macro could be a



From Page 13

standard letter. The names and addresses for each "individualised" letter could then be added from the parameter file that the Database has created from the data file.

All in all, it's an excellent piece of software. Powerful and flexible it's simple to learn and easy to use, though I advise practising on short, uncomplicated files before you computerise your firm's records.

While it's always easy to think of a few more options you could do with, I think that the Database has all the facilities that any normal user, home or business, would need.

What I particularly like is the ease of use, single key entry of commands being the order of the day. Given this and a manual that's well above the usual standard I have no hesitation in recommending it to people who have little or no previous experience of databases or even micros.

Having said that though, with a piece of software with as many options as the Electron Database it does take time and practice before you'll be an expert in its use.

So to sum up, it's a powerful, practical, straightforward piece of software that turns the Electron into a serious tool, suitable for both home and small business use.

Surprisingly easy to use, it has a depth and scope that only becomes apparent with familiarity and practice.

And, to answer the question posed in the first paragraph, it fits the first definition, not the second. If you keep written records, maintaining them on Database files should make your life a lot easier. Recommended.

JOYSTICKS— THE COMPLETE SOLUTION

DELTA 3B SINGLE—BBC B or ELECTRON PLUS 1 £12.00
A single joystick that in some ways can act as two. The custom made special "low noise" potentiometers are wired so that it will work as a left hand or right hand joystick. It can even run some programs written for two joysticks and has the fire buttons of both.

DELTA 3B TWIN—BBC B or ELECTRON PLUS 1 £19.95
A direct but improved alternative for the original ACORN joysticks, with 2 joysticks wired to one plug. As with all our joysticks they have the fast action sprung to centre return of the steel shafted nylon covered joystick. The light action makes them ideal to hold and the 3 fire buttons allow left or right-handed use.

Available from your dealer or direct from us

VISA

Voltmace Limited

Park Drive
Baldock
Herts
SG7 6EW
Telephone (0462) 894410

FRUIT WORM

By CARL DUNKLEY

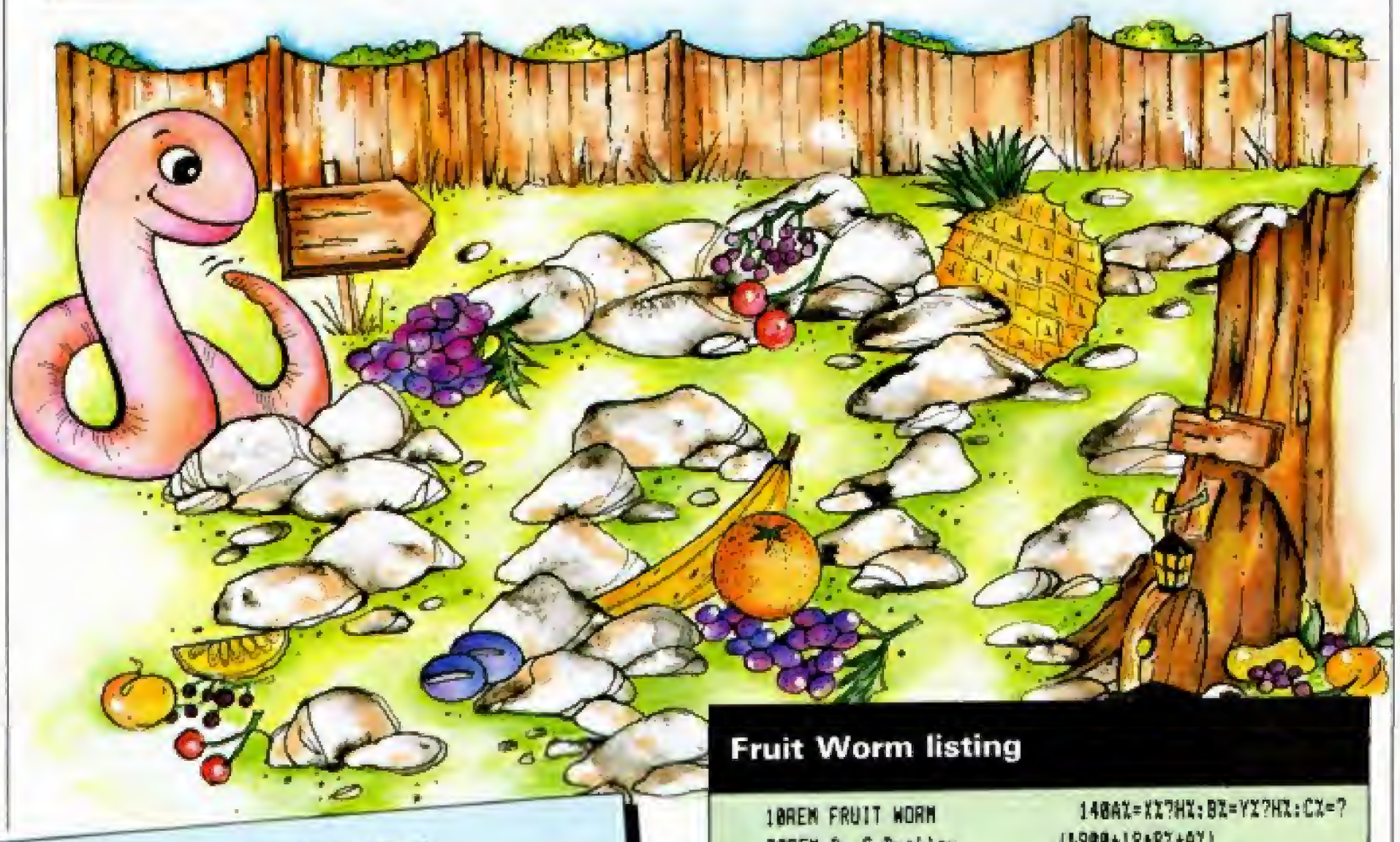
CONSERVE nature – help this poor worm round the screen as it eats fruit and avoids rocks, the outer fence and its own tail.

A bonus is awarded at the end of each frame and an extra worm is awarded for every 50000 points scored.

The game should pose no problems to enter, but it may be advisable to save it before running as it contains a lot of machine code.

All the instructions and movement keys are contained in the game itself.

So wriggle off . . .



PROCEDURES

pause	Wait for key depression.
erase(I)	Erase snake and increment score by I*length.
dead	Dead subroutine.
bonus	Bonus subroutine.
Print(A,B,B$)	Print at location A,B the string B$.
table	Display high score table.
swap	Adjust high score table.
exchange	Swap names and scores.
input	Input name.
initialise	Set up envelopes and arrays.
titles	Instructions.
wait	Play a tune while waiting for a key press.
screen	Draw screen.
locate(A,B)	Find an empty screen position.
characters	Redefine characters.
colours	Define colours.
black	Blank out display.
assemble	Assemble machine code.

Fruit Worm listing

```
  10REM FRUIT WORM
  20REM By C.Dunkley
  30REM (c) Electron User
  40MODE5
  50PROCinitialise
  60REPEAT
  70PROCtitles
  80REPEAT
  90PROCscreen
 100REPEAT
 110FORO%=0TO9%:NEXT:E%=0:
H%=H%+1:CALLkeys:IFK%=145PR
OCpause
 120IFH%=60H%=2:X%?1=X%?59
:Y%?1=Y%?59
 130X%?H%=?(X%+H%-1)-(D%=1
51)+(D%=155):Y%?H%=?(Y%+H%-
1)+(D%=182)-(D%=183)
 140A%=X%?H%:B%=Y%?H%:C%=?
(&900+18+B%+A%)
 150IFC%=10RA%=0ORA%=190AB
%=3ORB%=30:I%=I%-2%:I%=-1:G
OTO220
 160IFC%=2Z%=Z%-1:E%=1:S%=
S%+100:SOUND1,2,10,4:PRINTT
AB(1,1)FNf(S%,6)
 170CALLhead:A%=?(X%+H%-1)
:B%=?(Y%+H%-1):CALLbody:?(&
900+18+B%+A%)=1
 180IFE%<>0 GOTO220
 190A%=X%?T%:B%=Y%?T%:CALL
erase:?(&900+18+B%+A%)=0
 200T%=T%+1:IFT%=60T%=2:X%
?1=X%?59:Y%?1=Y%?59
```

→

Fruit Worm listing

From Page 15

```
  210AX=XX?TX:BX=YX?TX:CALL
body:?(&900+10*BX+AX)=1
  220UNTILZX<1
  230IFZX PROCdead ELSEPROC
bonus
  240UNTILLX=0
  250COLOUR129:COLOUR3
  260PROCprint(350,650,"GAM
E  OVER")
  270TIME=0:REPEAT UNTILTIM
E>275
  280RESTORE4060
  290FORNX=1TO8:READA,B:SOU
ND1,4,A,B:NEXT
  300TIME=0:REPEAT UNTILTIM
E>475
  310PROCtable
  320UNTIL0
  330END
  340:
  350DEFFNf(AX,BX):COLOUR2:
PRINTRIGHT$("00000"+STR$AX,
BX);:COLOUR3
  360IFSX>=PXANDLX<7SOUND1,
2,50,6:LX=LX+1:PX=PX+50000:
OX=AX:WX=BX:BX=2:AX=LX+6:CA
LLup:AX=OX:BX=WX:="" ELSE="
  370:
  380DEFPROCpause
  390CX=138
  400*fx21
  410REPEATUNTIL?&EC=0 AND?
&ED=0
  420REPEAT
  430PROCwait
  440CALLkeys
  450UNTILKX<>0
  460*fx15
  470ENDPROC
  480:
  490DEFPROCerase(I)
  500FORNX=1TOIX
  510AX=?(XX+TX):BX=?(YX+TX
)
  520TX=TX+1:IFTX=68TX=2:XX
?1=XX?59:YX?1=YX?59
  530IFZX CALLsplat
  540SOUND0,-1,5,4
  550CALLerase
  560SX=SX+I
  570PRINTTAB(1,1);FNf(SX,6
)
  580FORN=1TO45:NEXT
  590SOUND1,0,0,1
  600FORN=1TO25:NEXT
  610NEXT
  620ENDPROC
  630:
  640DEFPROCdead
  650SOUND0,-1,5,8
  660PROCerase(10)
  670COLOUR2
  680PROCprint(286,850,"YOU
'RE DEAD!")
  690LX=LX-1
  700RESTORE4000
  710FORNX=1TO11
  720READAX,BX
  730SOUND1,2,AX,BX
  740SOUND1,0,0,1
  750NEXT
  760FORN=1TO800:NEXT
  770ENDPROC
  780:
  790DEFPROCbonus
  800PROCerase(50)
  810QX=QX+4+(QX<>0)
  820RX=RX+4:IFRX=60RX=32
  830B=FRX*200
  840PROCprint(260,832,"NEI
T PATTERN")
  850PRINTTAB(4,13);
  860$&110="BONUS:":CALLdou
ble
  870COLOUR2
  880$&110=RIGHT$("00000"+S
TR$B,6):CALLdouble
  890FORN=1TO330:NEXT
  900FORNX=1TOB/100
  910SX=SX+100:B=B-100
  920PRINTTAB(1,1)FNf(SX,6)
TAB(10,13);
  930COLOUR2:SOUND0,-1,5,3
  940$&110=RIGHT$("00000"+S
TR$B,6):CALLdouble
  950SOUND1,0,0,0
  960NEXT
  970FRX=FRX+1
  980RESTORE4790
  990FORNX=1TO25
 1000READAX,BX
 1010IFAX=0 VX=0 ELSEVX=2
 1020SOUND1,VX,AX,BX
 1030SOUND1,0,0,1
 1040NEXT
 1050FORN=1TO700:NEXT
 1060ENDPROC
 1070:
 1080DEFPROCprint(A,B,B$)
 1090VDU5
 1100FORNX=1TO3
 1110GCOL0,2-(NX=1)+(NX=2)
 1120MOVEA+6*NX,B+6*NX
 1130$&110=B$:CALLdouble
 1140NEXT
 1150VDU4
 1160ENDPROC
 1170:
 1180DEFPROCtable
 1190PROCblack
 1200line=0
 1210COLOUR128:CLS
 1220COLOUR1:COLOUR131
 1230PROCprint(270,960,"Hig
h Scores")
 1240COLOUR128
 1250IFSX>score%(9) PROCswa
p
 1260FORNX=1TO9
 1270X=NX*2+4
 1280COLOUR1
 1290PRINTTAB(0,X);NX;")";T
AB(8,X);"...."
 1300COLOUR2
 1310PRINTTAB(2,X);name$(NX
);
 1320COLOUR3
 1330PRINTTAB(13,X);RIGHT$(
"   "+STR$score%(NX),6)
 1340NEXT
 1350PROCcolours
 1360IFline<>0 PROCinput
 1370COLOUR1:COLOUR131
 1380PROCprint(270,120,"Pre
ss any key")
 1390CX=138
 1400*fx15
 1410REPEAT
 1420PROCwait
 1430UNTILCX=45ORINKEY$0<>"
"
 1440ENDPROC
 1450:
 1460DEFPROCswap
 1470score%(9)=SX:name$(9)=
"":line=9
 1480FORLX=8TO1STEP-1
 1490IFscore%(LX+1)>score%(
LX) PROCexchange
 1500NEXT
 1510ENDPROC
 1520:
 1530DEFPROCexchange
 1540s=score%(LX):score%(LX
)=score%(LX+1):score%(LX+1)
=s
 1550s$=name$(LX):name$(LX)
=name$(LX+1):name$(LX+1)=s$
 1560line=LX
 1570ENDPROC
 1580:
 1590DEFPROCinput
 1600*fx15
 1610PRINTTAB(5,25);"Please
 enter"''TAB(6)"your name."
 1620*fx202 40
 1630VDU23,1,1,0,0;0;0;
 1640Y=line*2+4
 1650PRINTTAB(2,Y);
 1660RESTORE4810
 1670FORNX=1TO18:READA,B:SO
UND1,-15,A,B:NEXT
 1680L=0:a$=""
 1690REPEAT
 1700PRINTTAB(L+2,Y);
 1710a=GET
 1720IFa>31 ANDa<127 ANDL<5
 L=L+1:VDUa:a$=a$+CHR$a
 1730IFa=127 ANDL>0 VDU127:
L=L-1:a$=LEFT$(a$,L)
 1740UNTILa=13
 1750VDU23,1,0;0;0;0;
 1760name$(line)=a$
 1770COLOUR2
 1780PRINTTAB(2,Y);a$
 1790*fx202
 1800PRINTTAB(5,25);SPC(90)
 1810ENDPROC
 1820:
 1830DEFPROCinitialise
 1840*fx11
 1850*fx15
 1860*fx229 1
 1870IF?&212=27 OSCLI"fx163
 128 1"
 1880ENVELOPE1,1,3,-90,1,1,
-2,-3,126,0,0,-126,126,126
 1890ENVELOPE2,1,1,-2,1,7,5
,12,126,0,0,-126,126,126
 1900ENVELOPE3,1,0,0,0,10,1
0,10,126,-4,-1,-4,126,0
 1910ENVELOPE4,2,0,0,0,0,0,
0,64,-7,0,0,126,0
 1920rnd=RND(-TIME)
 1930DIMMX700,XX60,YX60,nam
e$(9),score%(9),fruit(3)
 1940score%(1)=340400:name$
(1)=" LSI"
 1950FORNX=2TO9
 1960score%(NX)=500*(21-NX)
 1970name$(NX)="Acorn"
 1980NEXT
 1990PROCassemble
 2000PROCcharacters
 2010ENDPROC
 2020:
 2030DEFPROCtitles
 2040VDU23,1,0;0;0;0;
 2050CLS
 2060SX=0:LX=3:RX=0:FRX=1:P
X=50000
 2070QX=248
 2080PROCblack
```

```
2090COLOUR1:COLOUR128
2100FORNX=1TO59
2110PRINT" C.Dunkley";
2120NEXT
2130PRINT" C.Dunkle";
2140COLOUR3
2150PRINTTAB(5,28);"C.Dunk
ley";
2160PROCprint(350,1000,"Fr
uit Wora")
2170PROCprint(40,70,"Sound
ON/OFF (S/Q)?")
2180VDU24,80;170;1236;910;
2190GCOL0,131:CLG
2200VDU28,1,25,18,3
2210COLOUR3:COLOUR130:CLS
2220PRINT'" Guide the wor
a around the screen,eating
:          and avoiding th
e rocks,outer fence and yo
ur own tail.";
2230PRINT'"A BONUS is awar
dedat the end of eachframe.
"
2240PRINT'"Extra worms are
   awarded for every 50,000
pts scored."
2250COLOUR1
2260PRINT'"   CONTROLS: A"
;TAB(12);"< >";SPC(16);"Z";
2270PRINT'"   P - Pause "
;:COLOUR3:PRINT"ON";
2280BX=7
2290FORNX=1TO3
2300AX=6+2*NX:CALLfruit(NX
)
2310NEXT
2320RESTORE4870
2330PROCcolours
2340FORNX=1TO10:READA,B:SO
UND1,-10,A,B:NEXT
2350*fx21
2360CX=133
2370REPEAT
2380PROCwait
2390A=INKEY0 OR32
2400UNTILA=113 ORA=115
2410*fx15
2420OSCLI"fx210 "+STR$~(A=
113)
2430VDU26
2440ENDPROC
2450:
2460DEFPROCwait
2470t=4
2480CX=CX+1
2490IFCX>188 CX=1:RESTORE4
820
2500IFCX>0 ANDCX<17 t=1
2510IFCX>46 ANDCX<67 t=2
2520IFCX>86 ANDCX<129 t=3
2530IFt=4 FORloop=1TO250:N
EXT:ENDPROC
2540READA,B
2550IFt=1 SOUND1,3,A,B:END
PROC
2560IFt=2 SOUND1,-15,A,B:E
NDPROC
2570IFt=3 ANDA=0 SOUND1,0,
A,B ELSESOUND1,-15,A,B
2580ENDPROC
2590:
2600DEFPROCscreen
2610COLOUR128:CLS
2620PROCblack
2630FORAX=1TO29
2640FORBX=1TO18
2650?(&900+BX+AX*18)=0
2660NEXT,
2670COLOUR129
2680PRINTTAB(19,31);" ";TA
B(0,0);STRING$(40," ");TAB(
0,30);STRING$(39," ");
2690COLOUR3
2700PRINTTAB(1,30);"FR:"TA
B(3,0);"SC"SPC(10);"HI"
2710PRINTTAB(1,1);FNf(SX,6
);SPC(6);FNf(scoreX(1),6)TA
B(4,30);FNf(FRX,2)
2720PRINTTAB(0,2);CHR$224;
STRING$(18,CHR$225);CHR$226
;
2730PRINTTAB(0,29);CHR$227
;STRING$(18,CHR$225);CHR$22
8;
2740FORNX=3TO28
2750PRINTTAB(0,NX);CHR$229
;TAB(19,NX);CHR$229;
2760NEXT
2770?&98E=1
2780?&98D=1
2790?&98F=1
2800?&9D0=1
2810?&9AC=1
2820AX=10:BX=10:CALLright
2830AX=9:CALLbody
2840FORNX=1TOLX
2850AX=NX+6
2860BX=2:CALLup
2870NEXT
2880PROCprint(375,50,"Frui
t Wora")
2890ZX=RX:IX=IX+2
2900FORNX=1TOZX/4
2910PROClocate(1,rock)
2920NEXT
2930FORNX=1TOZX+8
2940PROClocate(2,fruit(RND
(3)))
2950NEXT
2960PROCcolours
2970RESTORE
2980FORNX=1TO17
2990READAX,BX
3000IFAX=0 VX=0 ELSEVX=2
3010SOUND1,VX,AX,BX
3020SOUND1,0,0,1
3030NEXT
3040DX=151:KX=0
3050HX=2:TX=1
3060XX?1=9:YX?1=10
3070XX?2=10:YX?2=10
3080?&98F=0
3090?&9D0=0
3100?&9AC=0
3110PROCpause
3120COLOUR129:COLOUR2
3130ENDPROC
3140:
3150DEFPROClocate(a,b)
3160REPEAT
3170AX=RND(18):BX=RND(26)+
3
3180UNTIL?(&900+BX*18+AX)=
0
3190?(&900+BX*18+AX)=a
3200CALLb
3210ENDPROC
3220:
3230DEFPROCcharacters
3240VDU23,224,0,1,3,6,12,2
5,51,102
3250VDU23,225,0,255,255,0,
0,255,255,0
3260VDU23,226,0,128,192,96
,48,152,204,102
3270VDU23,227,102,51,25,12
,6,3,1,0
3280VDU23,228,102,204,152,
48,96,192,128,0
3290VDU23,229,102,102,102,
102,102,102,102,102
3300ENDPROC
3310:
3320DEFPROCcolours
3330VDU19,0,2;0;19,1,1;0;1
9,2,3;0;19,3,0;0;
3340ENDPROC
3350:
3360DEFPROCblack
3370FORNX=0TO3:VDU19,NX,0;
0;:NEXT
3380ENDPROC
3390:
3400DEFPROCassemble
3410FORNX=0TO2STEP2
3420PX=WX:[OPTNX
3430.erase
3440LDA#erase_data MOD256:
STA&72
3450LDA#erase_data DIV256:
STA&73
3460JMPprint
3470.rock
3480LDA#rock_data MOD256:S
TA&72
3490LDA#rock_data DIV256:S
TA&73
3500JMPprint
3510.splat
3520LDA#splat_data MOD256:
STA&72
3530LDA#splat_data DIV256:
STA&73
3540JMPprint
3550.fruit(1)
3560LDA#fruit1_data MOD256
:STA&72
3570LDA#fruit1_data DIV256
:STA&73
3580JMPprint
3590.fruit(2)
3600LDA#fruit2_data MOD256
:STA&72
3610LDA#fruit2_data DIV256
:STA&73
3620JMPprint
3630.fruit(3)
3640LDA#fruit3_data MOD256
:STA&72
```

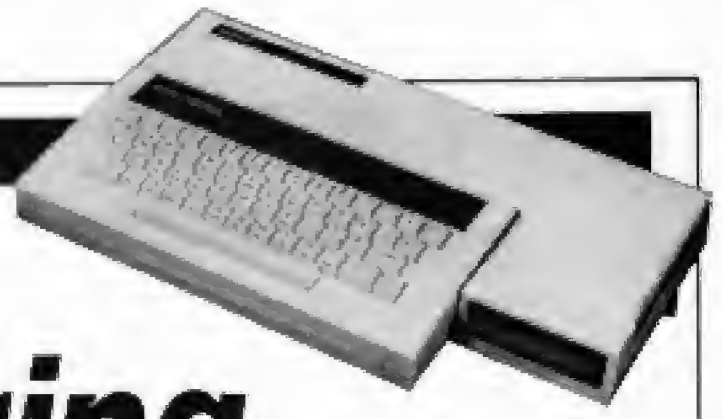
→

Fruit Worm listing

From Page 17

```
 3650LDA#fruit3_data DIV256
:STA&73
 3660JMPprint
 3670.left
 3680LDA#left_data MOD256:S
TA&72
 3690LDA#left_data DIV256:S
TA&73
 3700JMPprint
 3710.right
 3720LDA#right_data MOD256:
STA&72
 3730LDA#right_data DIV256:
STA&73
 3740JMPprint
 3750.up
 3760LDA#up_data MOD256:STA
&72
 3770LDA#up_data DIV256:STA
&73
 3780JMPprint
 3790.down
 3800LDA#down_data MOD256:S
TA&72
 3810LDA#down_data DIV256:S
TA&73
 3820JMPprint
 3830.body
 3840LDA#body_data MOD256:S
TA&72
 3850LDA#body_data DIV256:S
TA&73
 3860JMPprint
 3870:
 3880.fruit1_data
 3890EQUD&03896600:EQUD&01B
30707:EQUD&0C088866:EQUD&08
8CC24A
 3900.fruit2_data
 3910EQUD&11001166:EQUD&062
D0622:EQUD&06CC0000:EQUD&00
000620
 3920.erase_data
 3930EQUD0:EQUD0:EQUD0:EQUD
0
 3940.splat_data
 3950EQUD&7D657FAA:EQUD&125
3DABB:EQUD&AABBCCDD:EQUD&11
002200
 3960.rock_data
 3970EQUD&31101010:EQUD&F0F
76371:EQUD&6CECC880:EQUD&C0
FC3E3E
 3980.fruit3_data
 3990EQUD&65329966:EQUD&335
4ADDA:EQUD&A24C9966:EQUD&CC
6AB553
 4000.body_data
 4010EQUD&DFAF5733:EQUD&336
7DFAF:EQUD&5DBB6ECC:EQUD&CC
AE5FBF
 4020.left_data
 4030EQUD&DF763311:EQUD&112
356AF:EQUD&5FBF6ECC:EQUD&CC
AE5F8F
 4040.right_data
 4050EQUD&AFDF6733:EQUD&335
7AFDF:EQUD&0FE6CC88:EQUD&88
4CA65F
 4060.up_data
 4070EQUD&FD673311:EQUD&336
7DFAF:EQUD&5BAE4C88:EQUD&CC
AE5F8F
 4080.down_data
 4090EQUD&DFAF5733:EQUD&112
357AD:EQUD&5FBF6ECC:EQUD&08
CC6EFB
 4100:
 4110.print
 4120LDA#HIMEM MOD256:STA&7
4
 4130LDA#HIMEM DIV256:STA&7
5
 4140LDA&404:ASLA:ASLA:TAX:
LDY#4
 4150.loop1
 4160TXA:CLC:ADC&74:STA&74:
BCCn1:INC&75
 4170.n1
 4180DEY:BNEloop1
 4190LDA&408:ASLA:ASLA:ASLA
:TAX:LDY#40
 4200.loop2
 4210TXA:CLC:ADC&74:STA&74:
BCCn2:INC&75
 4220.n2
 4230DEY:BNEloop2
 4240LDY#0
 4250.loopP
 4260LDA(&72),Y:STA(&74),Y:
INY:CPY#16:BNEloopP:RTS
 4270:
 4280.keys
 4290LDA&EC:LDX&ED
 4300BEQcontkeys
 4310TXA
 4320.contkeys
 4330STA&42C
 4340CMP#151:BEQyes
 4350CMP#155:BEQyes
 4360CMP#182:BEQyes
 4370CMP#183:BNEhoaekeys
 4380.yes
 4390STA&410
 4400.hoaekeys
 4410RTS
 4420:
 4430.head
 4440LDA&410
 4450CMP#155:BNEnotleft:JMP
left
 4460.notleft
 4470CMP#151:BNEnotright:JM
Pright
 4480.notright
 4490CMP#182:BNEnotup:JMPup
 4500.notup
 4510CMP#183:BNEnotdown:JMP
down
 4520.notdown
 4530RTS
 4540:
 4550.double
 4560LDA#0:PHA
 4570.loopD1
 4580PLA:TAX:LDA&110,X:CMP#
13:BEQfini
 4590STA&80
 4600INX:TXA:PHA
 4610LDX#&80:LDY#0:LDA#10:J
SR&FFF1
 4620LDA#23:JSR&FFEE:LDA#25
5:JSR&FFEE
 4630LDY#1
 4640.loopD2
 4650LDA&80,Y:JSR&FFEE:JSR&
FFEE
 4660INY:CPY#5:BNEloopD2
 4670LDA#255:JSR&FFEE:LDA#0
:JSR&FFEE:LDA#10:JSR&FFEE
 4680LDA#23:JSR&FFEE:LDA#25
5:JSR&FFEE
 4690.loopD3
 4700LDA&80,Y:JSR&FFEE:JSR&
FFEE
 4710INY:CPY#9:BNEloopD3
 4720LDA#255:JSR&FFEE:LDA#1
1:JSR&FFEE
 4730JMPloopD1
 4740.fini RTS
 4750):NEXT
 4760ENDPROC
 4770:
 4780DATA136,2,128,2,136,12
,128,2,120,2,116,2,108,2,10
4,8,100,16,0,2,136,2,128,2,
136,12,116,6,120,6,104,6,10
8,8
 4790DATA100,1,116,1,100,1,
100,1,96,1,88,1,80,2,88,1,9
6,1,88,1,80,1,76,1,68,1,68,
2,100,1,116,1,100,1,100,1,9
6,1,80,1,80,1,96,1,68,2,76,
4,80,4
 4800DATA68,8,68,6,68,2,68,
8,80,6,76,2,76,4,68,4,68,6,
64,2,68,8
 4810DATA101,5,121,3,137,5,
121,2,129,2,121,2,129,2,121
,4,117,3,101,5,121,3,137,5,
121,2,129,2,121,2,129,2,121
,4,117,3
 4820DATA121,2,141,8,141,8,
149,8,149,8,169,14,157,2,14
1,6,141,2,157,6,141,2,129,8
,161,16,149,6,137,2,141,16
 4830DATA53,5,69,5,81,6,89,
3,81,5,69,5,53,5,69,5,81,6,
89,3,81,5,69,5,53,5,69,5,81
,15,89,15,73,6,69,3,73,5,61
,10
 4840DATA101,5,121,5,0,0,12
1,5,117,3,109,2,117,4,121,5
,101,4,0,1,101,4,0,1,101,4,
121,5,0,0,121,5,117,2,109,2
,117,4,121,5,101,4,0,1,101,
4,0,1
 4850DATA101,4,89,5,0,0,89,
5,81,5,0,0,81,5,73,6,0,7,10
1,5,89,5,0,0,89,5,81,5,0,0,
81,5,73,6,0,10
 4860DATA101,8,81,8,101,16,
117,4,121,4,109,4,117,4,101
,16
 4870DATA129,9,117,4,121,4,
129,9,101,9,121,4,129,4,137
,4,145,4,149,4
```

This listing is included in this month's cassette tape offer. See order form on Page 61.



Transferring from tape to disc

(but leaving no loopholes for pirates)

THIS month we're going to sort out one of the major problems that face disc owners – how to transfer tape programs to disc.

But we aren't going to be hacking into commercial software. The programs listed here won't cope with even the simplest protection system.

First we'll see how to copy the programs to disc, then how to get them running.

The initial part will apply to both Plus 3 and Cumana owners, but since the Cumana system doesn't take up any of your precious memory, getting the software to run isn't such a big problem.

Plus 3 owners may need a downloader, as we'll see later.

It doesn't matter whether the software on tape is a Basic listing, machine code, spooled Ascii text or a data file – they are all treated in the same way.

Program I will transfer any unprotected file to disc. If you have a Plus 3 then just run it. Cumana owners should change line 140 to:

```
140 *DISC
```

The ADFS in the Plus 3 requires a fair chunk of RAM as work space. Consequently you'll find that some programs completely fill the memory available and start loading into the memory reserved for the screen.

The program changes to Mode 6 and defines a text window, restricting text output to the lower half of the screen.

This enables the top half to be used as a buffer for really long programs such as adventures.

Since Program I is very short the memory from &2000 is free, so files are forced to load to &2000.

When the file has loaded its name, load, length and execution addresses are found from the header stored &3B2.

Figure I shows the header format.

The file is then saved to disc and the program looks for the next file on tape.

Once all the files are on disc you can try to run them. Some will run without modification of course, but you're bound to have problems with others.

We'll look at Basic programs first, as these are the easiest to sort out.

The most common error reports when the programs are run with a Plus 3 attached is "Bad mode" or "No room".

If the program loads without spilling over into the screen memory then all that is needed is a downloader – a routine which will move the program down in memory and reset PAGE to &E00. This is where PAGE is on the unexpanded Electron or Electron with a Cumana interface.

Program II is a procedure to relocate a Basic program. It should be added to the end of the program and the following line added to the start:

```
10IF PAGE)&E00
PROCrelocate:END
```

Function key 0 is defined to move the program down to &E00 and automatically run it. You don't need to press the function key as *FX138,0,128 inserts its code into the keyboard buffer for you.

A machine code relocator is used to relocate machine code programs and Basic programs that spill over into the screen memory when loading.

Program III is the machine code equivalent of Program II. The length, load and execution addresses should be placed in line 30. If you haven't got a Plus I leave out lines 150 and 160.

In the last article we saw that each directory on the Plus 3 disc contains an entry for every file saved in it. Each of these entries contains the information we require for line 30 in Program III.

Getting that information is quite simple, we use *INFO.

```
*INFO name
```

prints the normal information supplied by *CAT plus the load, execution, length and disc addresses in this order across the screen.

Figure II is a typical output showing *CAT followed by *INFO.

Set the variables *load*, *exec* and *len* according to the values

```
  10 REM PROGRAM I
  20 REM Tape -> Disc
  30 MODE 6
  40 VDU 28,0,24,39,15
  50 HIMEM=&72BB
  60 *FX16
  70 *TAPE
  80 *LOAD "" 2000
  90 A%=0
 100 REPEAT
 110 n$=n$+CHR$(A%?&3B2)
 120 A%=A%+1
 130 UNTIL A%=11 OR A%?&3B
2=0
 140 *ADFS
 150 n$="SAVE "+n$+" 2000
+"+STR$~(?&3C6+256+(!&3C8 A
ND &FFFF))+" "+STR$~(!&3C2)
+" "+STR$~(!&3BE)
 160 PRINT'"*";n$'
 170 OSCLI n$
 180 CLEAR
 190 GOTO 70
```

Program I

```
10000 DEF PROCrelocate
10010 VDU 21:*FX200,2
10020 *KEY0 "*TAPE|MO%=PAGE
-&E00:FORI%=PAGE TO TOP STE
P4:!(I%-O%)=!I%:NEXT:?(TOP-
O%)=255|MPAGE=&E00|MOLD|MRU
N|F|M"
10030 *FX21,0
10040 *FX138,0,128
10050 ENDPROC
```

Program II

&3B2	10 bytes	filename
&3BE	4 bytes	load address
&3C2	4 bytes	execution address
&3C6	2 bytes	block number
&3C8	2 bytes	length of block

Figure I: Cassette header information.

```
  10 REM PROGRAM III
  20 MODE 6:VDU 23,1,0;0;0
;0;
  30 load=&E00:exec=&2000:
len=&4321
  40 FOR N%=0 TO 2 STEP 2
  50 P%=&7F40
  60 [OPT N%
  70 .a LDA &2000,.b STA l
oad \relocate
  80 INC b+1:BNE d:INC b+2
  90 .d INC a+1:BNE c:INC
a+2
 100 .c LDA #&21+len DIV25
6:CMP a+2:BCS a
 110 JMP exec
 120 ]:NEXT
 130 *LOAD MCode 2000
 140 *TAPE
 150 *FX163,128,1
 160 ?&212=&D6:?&213=&F1:?
&2AC=0
 170 CALL &7F40
```

Program III

From Page 19

printed for your program.

The first four digits of each number aren't important so in the example in Figure II you would set these to &1D00, &8023 and &0130 respectively. The name should be placed in line 130.

When the loader is run the file is *LOADed to &2000. The tape filing system is selected, the Plus 1 switched off and the program relocated.

If the file is Basic then replace line 110 with:

```
110 RTS
```

and add:

```
145 PAGE=&E00
```

When the program has been relocated type OLD and RUN.

As it stands Program III will jump to the execution address to run the program. If you want to *LOAD the file and not *RUN it change line 110 to RTS.

Now you have your software running on disc we'll set up a !BOOT file automatically to load and run the menu creator we developed in last month's issue of *The Electron User*. You'll need *BUILD for this.

*BUILD is a utility on the Welcome disc in the Library directory. It's always best to copy this directory on to all your discs.

Copying is easy. Place the Welcome disc in the drive, hold down Ctrl+A and press Break.

Enter *DIRCOPY and follow the instructions, entering $.Library as the name of the source and destination directories.

The drive is 0, and answer Y when it asks if you want to overwrite locked files and if you want prompts.

Once you've copied Library and have *BUILD on the disc with your programs you can create a !BOOT file. After Ctrl+A+Break, type:

```
*BUILD !BOOT
```

and you'll see a line number appear. Now type:

```
CHAIN "Menu"
```

and press Return. You'll get a second line number and you could enter more instructions if necessary, but this is all we need, so press Escape to end.

If you *CAT the disc you'll see the !BOOT file. We now have to tell the Electron what to do with this file so enter:

```
*OPT4,3
```

This tells it to *EXEC the file.

When Shift is held down and Break pressed the !BOOT file will be *EXECed. This says CHAIN"Menu" so the menu is loaded and run.

The other *OPTions are:

0 – do nothing.
1 – *LOAD
2 – *RUN

So you could write a small machine code routine if you wanted and *RUN it or *LOAD it.

If you've followed the instructions you should now have a disc containing several of your programs. The menu can be run by holding down Shift and pressing Break.

Putting your software on to disc isn't that difficult once you get the hang of it.

Most programs just need the downloader, and it's only the odd one that needs the machine code. I think you'll find that it's worth the effort.

```
>*CAT
Disc . . . . .5                 (43)
Drive:0                         Option 03 (Exec)
Dir. $                          Lib. Library

!BOOT       WR  (11)    Database    DLR (15)
Library     DLR (02)    Menu        WR  (41)
Plus3       DLR (77)    Procedures  DLR (03)
Sprites     DLR (55)    Tape/Disc   WR  (43)
Toolkit     DLR (04)    Viewfiles   DLR (09)
ZYSysHelp   L   (01)
>
>
>*INFO Tape/Disc
Tape/Disc   WR  (43)    FFFF1D00    FFFF8023
00000130    00014D
```

Figure II

QUAL-SOFT **THOUGHTWARE** **Sports simulations**

"I find the packaging of this game beaten by only Revs and Elite, and the contents of the game beaten by none". Jason Sinclair, Leeds.

Qual-Soft comment: We are pleased to be compared with these excellent games and would point out that we are also **£5 CHEAPER THAN EITHER!**

In 1966 Alf Ramsey proved that English club soccer players, with intelligent management, could not only dominate European club football, but could take on, and beat the rest of the world at International level. Could you do the same in . . . .

TAPE 1 QUALIFIERS **MEXICO '86** **TAPE 2 FINALS**

A WORLD CUP MANAGEMENT SIMULATION

Summer 1984 and English International football is at its lowest ebb. We have failed to qualify for the European Nations Cup, and had a string of very poor International results. In a few months we will set out on the '86 World Cup qualifying trail. You have been given the most important job of restoring English pride in their football. You have a match in Paris, the USSR at Wembley, and a South American tour, to assemble a team, first to qualify, and then to beat the world's best in Mexico.

TAPE 1 (Qualifiers)

- Current squad of 16 players + 20 user defined players.
- Friendlies in Paris, at Wembley + South American tour
- ANY team formation you choose. 2 from 5 substitutes.
- In match tactics: any no. of individual player adjustments
- Your qualification group: full results and table.

TAPE 2 (Finals)

- Choose a 20 man squad to take to the finals.
- Group of 4 prelims. 16 to final knockout comp.
- Extra Time, PENALTY SHOOT-OUTS, where relevant
- Formation and strength information on opposition
- 2 from 9 substitutes (the FA tells us so).

ENGLAND'S GAMES: FULL PITCH, 22 MAN, 3D GRAPHICS & SOUND EFFECTS

QUAL-SOFT comments: With 5 levels of play, 12 depths of sophistication, and "fun" graphics, this game can be enjoyed by an 8 year old youngster as a "fun" game, and by the most sophisticated as a tactical/strategy challenge of the highest order

PACKAGE: Tape 1 plus Tape 2 plus 20 Page Manual in "Video Cassette" style pack. Only **£9.95** *(57K RAM usage. Some would call this a MEGAGAME)*

QUALSOFT GUARANTEE: Sent by 1ST CLASS POST on day ordered with P.O., Cheque, Access, payment is received.

QUAL-SOFT, Tel: 0438 721936
Dept. EU,
18 Hazelmere Rd.,
Stevenage, Herts SG2 8RX.

Please supply:
MEXICO '86
Electron ☐
BBC 'B' ☐

Name: .....
Address: .....
.....
Access No. (if applicable) .....

Software Surgery

THE COLUMN THAT TAKES A LOOK INSIDE THE LATEST RELEASES

Let down by the gremlins

Gremlins
Adventure International

I SUPPOSE I'm the wrong person to review Gremlins, for I haven't seen the film and I can't finish the adventure – despite the hint sheet that came with it!

The promotional material says the game is based on the film so I suspect that anyone who has seen it will have a decided advantage.

The object of the exercise is to kill all the gremlins. To encourage you, your mission starts in your bedroom with a gremlin throwing darts at you.

Unless you are quackers you make a quick exit into the living room, where you'll find a gremlin too, but he soon gets the point.

A quick visit to the kitchen will probably find you as baffled as me, so it's off out through the front door and into the driveway.

A vehicle in the garage will have you ploughed under if you aren't careful and then you are off down the road to explore the town.

The cinema is showing Gremlins (wouldn't you know it!) and the real McCoy are as interested in the ending as you are.

The petrol station is a useful hiding place, and the department store is well worth a look but don't let these new-fangled gizmos loose you.

I must admit that I felt there was something different about this adventure all the time I was playing it.

But somehow the atmosphere just didn't seem to be there, and I was left feeling slightly disappointed.

I haven't finished the game, and to be honest I don't think it generates enough interest to make me want to.

Overall, a capable enough adventure that is not quite up to the standard of the others from Adventure International.

Merlin

Harrier is a hot one

Strike Force Harrier
Mirrorsoft

IF you've ever wondered what it would be like to fly one of the world's best combat planes then took a look at Strike Force Harrier.

Mirrorsoft would probably agree that it's not quite as good as the real thing, but then it's about five million pounds cheaper!

The display is quite impressive, with excellent graphics.

Your instruments cover the lower half of the screen and consist of a map and radar, thrust and fuel gauges and the status of the undercarriage, flaps and brakes.

The windscreen covers the top half of the screen. This is where the action takes place. Through it you can see the ground, horizon, sky and clouds, plus a few more instruments.

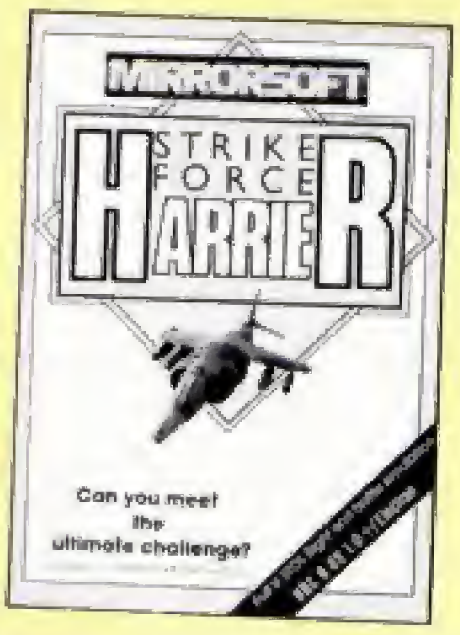
Flying the Harrier is fairly easy. Within 10 minutes I was looping the loop and doing barrel rolls.

This is only a small part of the game though. The plane is armed with bombs, missiles and cannons to defend yourself from surface to air missiles, anti-aircraft fire and enemy aircraft (MIG 23s).

Your mission is to destroy the enemy HQ 500 miles from your starting position. This is achieved by blasting enemy tanks on the ground with your cannons and bombs.

A ground site can then be set up and your own forces moved up by an airborne drop at a speed of around 600 knots.

Each new base needs defending from tanks as you



attempt to set up the next.

Strike Force Harrier is more than a single flight simulator, it's a battlefield simulation.

It requires a knowledge of ground attack techniques and skill in air to air combat.

The 27 page manual supplied describes these tactics in detail, along with an outline of your mission and tips on flying.

There are several different levels of difficulty, including a practice mode in which you aren't attacked.

After a bit of practice you can try your hand at combat. Far from easy this – it takes a long time to master.

If you're after something more than a flight simulator then Harrier is well worth considering.

The addition of ground and air combat makes this one of the best games of its type on the Electron.

Roland Waddilove

Easier than some sheets

Viewsheet – ROM Cartridge
Acornsoft

ELECTRONIC spreadsheets have always had a reputation of being difficult to use, and I'm sure that this reputation, not entirely unjustified, has been acquired largely through unfamiliarity.

At first glance Viewsheet looks to be incredibly complex, coming as it does with a 143 page manual and a huge range of facilities.

In fact it is straightforward to use, though the complexities may take a considerable time to master.

The manual guides you into the subject gently, showing that it is very easy to set up simple sheets such as the Magic Square, and gradually builds up to the more powerful commands.

On my first encounter with Viewsheet I was astonished by just how powerful these were, and could soon see why spreadsheets are considered the best way of manipulating data, numbers and calcul-

→

From Page 21

ations.

The program was designed and written by Protechnic, the company responsible for View, with which it is compatible.

Coming on a 16k ROM, it is switched in instead of Basic (by *SHEET) and so uses little of your precious RAM.

In common with View it is key rather than menu driven, the Escape key toggling between command and sheet modes.

The sheet has a nominal 255 by 255 size, and is best used in Mode 3.

At the top of the screen is a permanent display of command information – your current position within the sheet, and the contents of that slot.

The current slot itself is in reverse video – the sheet cursor. This can be moved around at will by using CapsLk/Func and another key for the appropriate direction.

Each slot can contain a label, formula or reference. For example, if in slot A1 you enter PI, you see 3.14159 appear.

Move to B1 and type 2*A1, and this is calculated and stored in B1 as 6.28319. B1 thus refers to A1, so go back to A1 and alter the contents to another number. In B1 that number x2 appears automatically.

This is the essential power of the spreadsheet. Any slot referring to A1 will be updated, and likewise any referring to B1 and so on, propagating any change across the whole sheet. Slots can also be accessed by naming the columns and rows, for example "JUNE" Week1" *0.15.

They can be filled by auto-entry (across or down), replication (across, down or both, and Absolute or Relative) and by editing existing contents.

As for calculations, a large range of functions are available including summing, conditional operators (including IF), pseudo-list functions such as MAX, MIN, AVERAGE, CHOOSE and LOOKUP (taking lists of slots as one of their arguments), and most of the Basic maths functions with the usual operator precedence.

Up to 10 windows can be defined to display information from all over the spreadsheet simultaneously. Printing is also by means of windows and drivers, and save/load windows commands mean that one large spreadsheet can be used with several sets of windows.

Other facilities include protection, insertion and deletion of rows and columns, forced recalculation, editing slot formats – decimal places, ranging right or left and altering the column width.

In addition Plus 3 users can transfer data between sheets using specially created files, and two or more spreadsheets can be linked, so overcoming memory limitations.

In short, this ROM is sufficiently powerful to provide almost any conceivable planning, modelling and forecasting that anyone could want from an Electron.

Many people don't realise the capabilities of spreadsheets, I'd advise them to go and try it!

My only reservations are the rather poor bar charts (using rows of asterisks), the lack of an index in the otherwise excellent manual, and the speed of some of the response times – none serious enough to prevent me from warmly recommending it.

At £29.95 it is now well priced, but beware: some shops are still selling it at double that. **Nick Rhodes**

Revision program

Chemistry
Letts Keyfacts
Revision Software

THIS chemistry package follows the standard format for Letts revision programs.

You get two cassettes crammed full of programs, along with a small booklet explaining how the software can be used all in a neat library case.

As usual there are ten programs. The first deals with atomic structure and bonding and is a Cloze activity.

The means aspiring O-level or CSE chemists must complete a piece of text by typing in the missing words.

They can choose whether to have only a few of the chosen words missing or the whole lot.

If they get stuck, pressing X enables them to see all of the text.

Activities like this are not only considered educationally sound, they should also be of real benefit to a revising pupil.

Program two moves on to the central pillar of inorganic chemistry, the periodic table. Students are expected to enter various elements on to a blank table.

Options include entering chemical symbols, atomic numbers or electron structures. The idea is good, but the presentation could have been improved. Nonetheless it is a useful revision program.

Formulae and equations come next. This is another very important topic and any fairly friendly help from a micro is a good thing.

This program considers various reactions and you have to balance the equation.

Alternatively, a list of chemicals is given and you decide if they are reactants or products.

The equation for the reaction is then given and you must balance it.

There follows a program called deductions. Here you are presented with a series of clues as to the identity of a substance.

Some clues are mathematical and calculator, pencil and paper will be needed. If you can't get the substance within five guesses you are told what it is.

The last program on tape one is concerned with electrolysis.

With a choice of molten electrolytes or solutions, you must identify the ions present and to which electrode they move. The ion equation must

Hacker on the levels

The Hacker
Firebird

THIS hacker really has little to do with hacking. However it makes an interesting and topical story line for the game.

It's actually a Manic Miner-type levels game. There are 12 screens and many different puzzles and obstacles to overcome.

Your objective is to gain access to the central computer games library. You are cast in the role of a small man who is able to pass through electrical circuits. It's here that the action takes place.

You must pass through your modem into the telephone network.

Then it's out of the mainframe's modem, into its buffer,

down its data bus and into the central processing unit. Then you can access the files.

On each screen are five floppy discs to collect and a time limit in which to do it. The time remaining when you've completed the screen is added to your score.

Any screen can be practised without having to start at the beginning and go through each one. This is a useful feature which I wish more games had.

The graphics are quite nice but the movement of the characters is fairly slow and they aren't very smooth. This spoils the game somewhat.

I should imagine it's much better on the BBC Micro with that little bit extra speed.

However having said that, Firebird software tends to be cheaper than most, so taking that into account it's a reasonable game.

Roland Waddilove

then be completed. Tape two begins with two related programs on apparatus. In the first of these you must put together the equipment needed to prepare the gas of your choice.

This is done by selecting the correct pieces from a collection and swapping them around. This is very fast, smooth and fun.

When completed, all other bits of apparatus are cleared from the screen and then you label your diagram and complete the equation for preparing the gas.

The second program gives you a wrong diagram. You identify the fault and label it.

The third program, acids, bases and salts, returns to the Cloze idea. There are four pages of text to complete.

Then in a separate section you must decide whether a reaction goes in the direction shown. You are also asked a few questions about it.

Pupils actually get marked on this section which is surely helpful when revising.

The third section introduces the concept of molarity and then asks you to predict titration results. It's a little complicated.

The package ends with two programs on organic chemistry. The first is a series of Cloze procedures, enlivened with diagrams to illustrate the principles involved.

Working through this program certainly reminded me of the organic chemistry I used to know.

The second program requires identification of isomers and homologues. I found this the weakest program on the tapes. There were just too many CHs on the screen for me.

The instructions suggest that this program is to test your powers of observation. Mine obviously are not good enough.

This is without doubt what it says – a revision package. Don't expect it to teach you if you know absolutely no chemistry, because you will end up very confused.

Despite the large amount of material here the price of £11.50 seems a bit steep.

It is, however, fit for its purpose and pupils taking chemistry at 16+ levels could find it useful. **Roger Frost**

Take a speedy tortoise

Caveman Capers
Icon Software

IF your idea of fun consists of tearing through the prehistoric wilderness standing on the back of a runaway tortoise, Caveman Capers is for you.

Press the Spacebar to start the game and the background begins to scroll from right to left.

Balancing precariously on the back of a turbo-charged tortoise, you must leap over the potholes as you career towards almost-certain disaster.

Some well-timed stabs at the Return key are sufficient to get you past these first obstacles.

Having been broken in gently, you will now be buzzed by pterodactyls while still jumping craters.

With the use of the Z and X keys to move our caveman hero short distances to the left and right, you can avoid nutting the birds.

Section three seems quite easy as there are no craters to jump – just a series of purple snakes hanging down from the treetops.

The problem with the snakes is that they are constantly bobbing up and down.

Should they be in the down position as you pass below, then it's PROCheadache for our hero and shellshock for his transport.

The next two levels are very similar.

The first involves using extended jumps to clear some toadstools and in the second you're jumping logs.

The next stage is the one that is giving me great difficulties at the moment. I'm having trouble with crabs.

One minute they are lying nice and quiet on the ground, the next thing you know they're doing flipping star jumps at about waist height!!

Icon has really gone to town on the graphics with this game as every character is drawn on a grand scale and in the brightest of colours.

One character who has not yet played a part in the game is a rather large dinosaur who keeps poking his head on to the screen and grinning mischievously. I'm sure he's up to no good.

Caveman Capers is an excellent game. It has got fast action, simple controls, and is just brimming with humour.

James Riddell

Venture on – it's well worth it

Adventureland
Adventure International

THIS was the first adventure ever to be released for a home computer, and if that isn't reason enough to buy it, let me state it is also an excellent one.

The object of your quest is to collect and store 13 treasures.

You start the game in a forest and though you aren't a social climber the view is disappointing.

The axe is soon found in a sunny meadow and when I tell you that Bunyon's first name was Paul you'll know where to visit when you try some magic.

You might bear this in mind when you visit the quicksand, a definite case of sink or swim!

The chiggers in the swamp will soon get under your skin, so protect it. The cypress tree is worth a visit but it is not necessary to do so to finish the game, though it will provide a clue.

Having been given the chop, your next problem is getting down to the main body of the game.

An examination of the room descriptions and a careful choice of wording is as essential here as it can be elsewhere.

The lamp and a safe place should now be found, though you might not yet be able to light the lamp.

Before you go back up you might consider playing the lead in Aladdin II. A score of 38 out of 100 should be your mark and a means of lighting the lamp will also be found underground, though don't go too far!

If you die you will go to limbo and you only have one move you can make here – a Hell of a choice!

By now you are well into the main body of the adventure, and though you will still have to return to the surface several times the answer to most of your problems lies underground.

Scott Adams rates this game at moderate level. I think it is more like beginners' level, especially when you compare it with some of his later games.

That is not to say that it doesn't have some fiendish problems though! What makes it easy is the level of help given.

The seasoned adventurer can expect to finish it in a couple of days, but the novice will find it the perfect introduction to the genre.

An excellent adventure that is worth having for its collectors' value alone. If you haven't got it, you should have!

Merlin

THE BEST BBC MICRO SOFTWARE

*** PRODUCED BY AN INDEPENDENT SOFTWARE HOUSE ***
*** TOP QUALITY MACHINE-CODE PROGRAMS ***

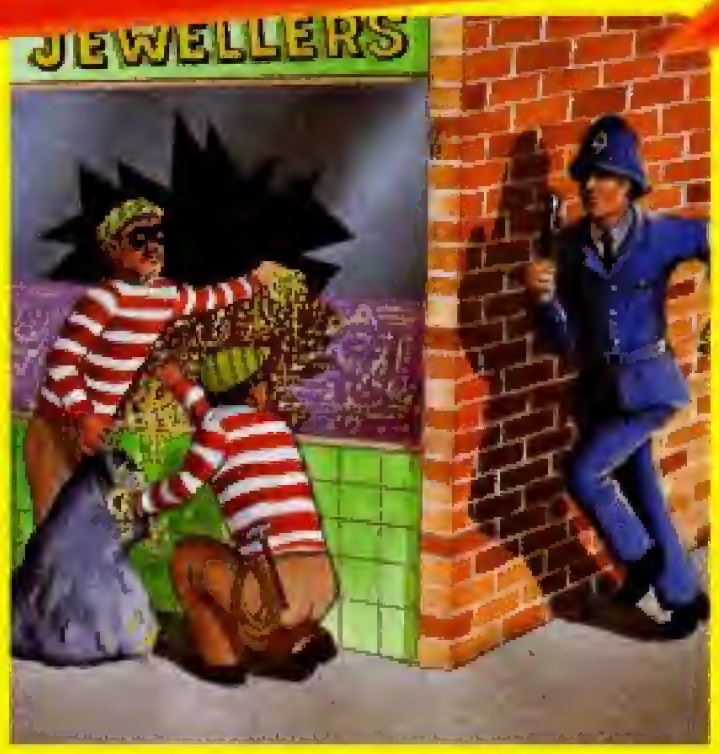
ACORN ELECTRON
BBC MICRO

SPECIAL OFFER!
Order both directly from us and deduct £1 from each.

OVERDRIVE (32K) **£7.95**

A highly-addictive multi-stage 3D race game. You steer your car left and right, accelerate and decelerate as the opposing cars weave about the road. There are five different stages including night, snow, desert and riverside scenes. To qualify for the next stage, you must finish in the top twelve. Incredible graphics give the impression that you really are taking part in the race. Highly recommended, and destined to become another top-seller for Superior Software.

BOTH TITLES ARE NOW AVAILABLE FOR THE B.B.C. MICRO AND THE ACORN ELECTRON.



SMASH AND GRAB (32K) **£7.95**

An excellent and original arcade-style game in which you take the role of a robber aiming to snatch bags of gold from the bank. A policeman is after you ... he is able to jump at you or squat down and try to hit you with his truncheon. You must also keep clear of the flying police cones and floating dustbin lids. There are three fascinating screens of action including play streets with bouncing balls, one-way streets, conveyor belts, traffic lights and police-boxes. A novel and amusing game.

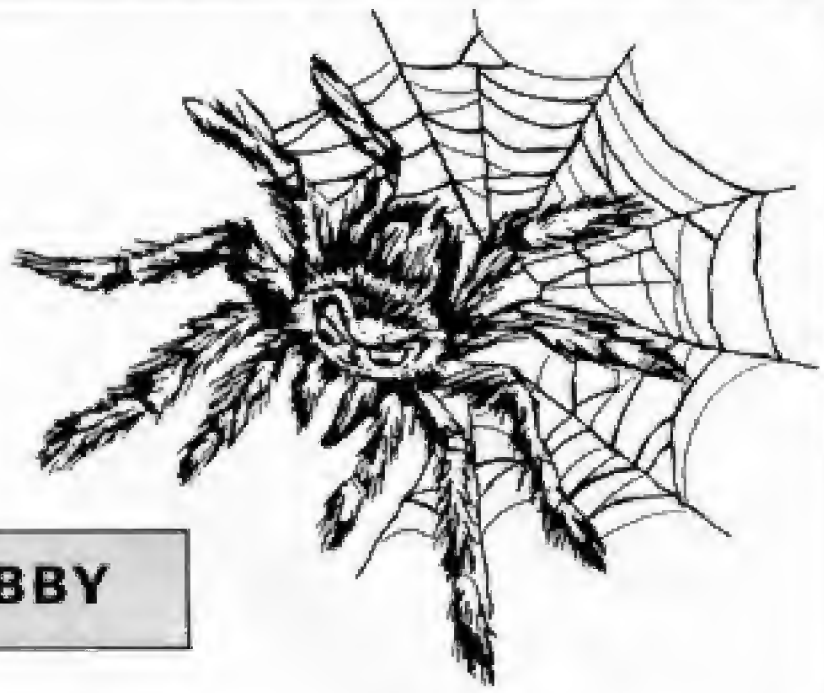
WE PAY UP TO 20% ROYALTIES FOR HIGH QUALITY BBC MICRO AND ELECTRON PROGRAMS.

SUPERIOR SOFTWARE LTD.
Dept. EU12, Regent House,
Skinner Lane, Leeds 7
Tel: 0532 459453

OUR GUARANTEE

(1) All our software is available before we advertise.
(2) All our software is despatched within 48 hours by first class post.
(3) In the unlikely event that any of our software fails to load, return your cassette to us and we will immediately send a replacement.

Searching IN STRings



By PETE BIBBY

THIS month we'll be exploring the INSTR function. This searches through one string to see if it contains another string.

If that sounds a bit strange, try entering the following lines to see INSTR in practice:

```
PRINT INSTR("BUS","B")
PRINT INSTR("BUS","U")
PRINT INSTR("BUS","S")
```

You'll find that they give you the numbers 1, 2 and 3 respectively.

What's happened is that the INSTR has searched through the first string in the brackets to find any occurrences of the second string. In the case of:

```
PRINT INSTR("BUS","B")
```

the Electron searches through the string "BUS" for any Bs that it might find.

When it finds a match the function returns the character position of the match in the first string. In this case the B matches the first character of BUS, so the number 1 is returned.

Similarly when we had the micro looking through BUS for U, the number 2 was returned as the match was found at the second letter of BUS. And I'll leave you to figure out why looking for S in BUS gives us 3.

You may be wondering what happens if the letter we're looking for isn't in the first string. What if there's no matching characters?

The answer as ever, is try and see. Use something like:

```
PRINT INSTR("BUS","X")
```

and you'll find that you get 0 for your pains. It's fairly logical:

If there is a match, the Electron returns the character position where the match was found.

If no match can be found, as in the case of BUS and X, then 0 is returned to tell you that there are no matches.

I leave it to you to explain why:

```
PRINT INSTR("BUS","u")
```

also gives you 0.

So far the examples inside the brackets have consisted of strings surrounded by quotes. They can also be string variables as:

```
search$="lookinhere"
target$="k"
PRINT INSTR(search$,
target$)
```

will show you. It returns 4, as the match is found at the fourth character of *search$*.

The target string doesn't have to be a single letter, it can be several characters long. Then INSTR will search for any occurrences of the whole target string inside the first string.

Try changing the target string with:

```
target$="in"
```

and then using:

```
PRINT INSTR(search$,
target$)
```

The result is 5. This match between the two strings begins at the fifth character of "lookinhere". Notice that the INSTR function only gives the position of the first character of the match.

Try changing *target$* with:

```
target$="kin"
```

and:

```
target$="this"
```

and explaining why you get 4 and 7 returned when you:

```
PRINT INSTR(search$,
target$)
```

You might also explore what happens when your target string is longer than your search string as in:

```
PRINT INSTR("abcd","abcde")
```

and:

```
PRINT INSTR("wx","xyz")
```

Now that you've got a reasonable grasp of the way INSTR works, let's see how it can be used in programs. Program I tests whether an input character is a vowel.

Lines 10 and 20 are fairly straightforward. All line 30 does is to check that *letter$* is just one character long.

It's a mugtrap that only allows you to enter what is asked for. If you input more than one letter it sends you straight back to the previous line.

Notice that this is one of the few times when I'll allow myself to use a GOTO. As it's only referring back to the previous line I don't think there's that much likelihood of the flow of control being messed up!

```
   10 REM PROGRAM I
   20 INPUT "Type in a lett
er " letter$
   30 IF LEN(letter$)>1 OR
LEN(letter$)=0 THEN GOTO 20
   40 IF INSTR("aeiou",lett
er$)=0 THEN PRINT "It's not
 a vowel" ELSE
```

Program I

Line 40 is where the action really is. The:

```
INSTR("aeiou",letter$)
```

checks to see if the letter you have typed in, held in the variable *letter$*, is in the string "aeiou". If it isn't then it's not a vowel and the string returns 0. The IF and ELSE combine to make sure that the appropriate message is displayed.

One thing about Program I is that it won't accept E as a vowel. The search string only has a lower case e in it, not upper case E. Can you alter the listing to allow the program to

From Page 25

detect upper case vowels?

Program II uses the principles of Program I's line 40 to use INSTR in constructing another mugtrap.

As you can see from line 20, all the program does is to print a trivial message. The interesting bit comes when it asks you if you want it to repeat the message.

The INPUT of line 30 asks you whether you want another go and prompts you to reply Y or N for yes or no.

Now there are several possible responses. The user may type N or n – which we want to stop the program – or Y or y when we want the program to continue.

More annoyingly, he may type in garbage which we want the micro to ignore. What the program does is resort to two mugtraps.

The first, in line 40, we've met before as line 30 of Program I. Line 50 makes up the second mugtrap.

It takes *letter$* (which must consist of only one letter as it got past the previous line) and sees if it has a match in "YyNn".

If it hasn't then the user hasn't typed in what was wanted and the GOTO gives him another chance.

By the time the program gets to line 60, *letter$* can only consist of one letter, either Y or y or N or n. This has to be the case or the mugtraps wouldn't have allowed the Electron to get this far.

Now INSTR is used to find a match with "Yy". If there is one, INSTR returns a value of 1 or 2 depending on which letter *letter$* contains. So if the result of:

```
INSTR("yy",letter$)
```

is not equal to 0, the program goes back to the beginning again.

If and when the program reaches line 70, *letter$* can only contain N or n. This is because the mugtraps and the check of line 60 will have filtered out all other possible inputs. Hence there's no need to use INSTR on *letter$*, the final message can be printed out immediately.

Program III does exactly the same job but in a slightly different way. The last three lines of the previous program have been replaced by just one line with two INSTRs. You shouldn't have too much difficulty in working out what's happening.

So, INSTR allows us to search for occurrences of one string in another, giving us the character position of the match.

As we've seen, this can be very useful but it does have its limits. To see what I mean, try entering:

```
PRINT INSTR("element","e")
```

which returns 1. Now this is all well and good, there is an e at the beginning of *element*. But there's also an e as its third character. And its fifth. What if we want these?

The answer lies in a feature of INSTR that we haven't looked at yet. Previously we've always started looking for a match at the first character of the search string. Hence:

```
PRINT INSTR("element","e")
```

gives the answer 1 as it finds the match at the first character of *element*. It ignores the rest.

However the other occurrences of e aren't lost to us. We can find their positions by adding another parameter inside the INSTR's brackets. To see what I mean try:

```
PRINT INSTR("element",
"e",2)
```

which returns 3 and:

```
PRINT INSTR("element",
"e",4)
```

which returns 5.

What has happened is that the number following the target string tells the micro where to start looking for the match. In the case of:

```
PRINT INSTR("element",
"e",2)
```

the figure 2 tells the micro to start looking for a match for e at the second character of *element*. This second character is, of course, the letter l.

The first match found after this is the e that's the third character of *element*, hence the figure 3 is returned. With:

```
PRINT INSTR("element",
"e",4)
```

the search starts at the fourth character of *element* m. The first match is found at the fifth letter of *element*, so 5 is displayed.

It might make things clearer if you realise that before, when we used:

```
PRINT INSTR("element","e")
```

the Electron took it as:

```
PRINT INSTR("element",
"e",1)
```

and started the search at the first character. So we haven't really added a new parameter to INSTR, we're changing one that the Electron had taken as being 1 since it hasn't heard anything to the contrary.

What happens if you tell the micro to search for a match and give it a starting place that doesn't exist? Try:

```
PRINT INSTR("DDD","D",4)
```

Here DDD is only three characters long, yet you're telling the micro to look for a match starting at its fourth character. The result is that the function returns 0.

You don't have to use a figure to specify where the search starts. A numeric variable will do the job, as:

```
search$="cocoa"
target$="co"
start=1
PRINT INSTR(search$,
target$,start)
```

shows. Try changing *start* to 2, 3 and 4 and see what happens.

That numeric variable at the end of INSTR looks very

start	INSTR()	count
1	1	1
2	3	2
3	3	3
4	5	4
5	5	5
6	0	5
7	0	5

Figure I (left): Shows double counting in a FOR . . . NEXT loop

Figure II (below): Shows how to avoid double counting

cycle	position	INSTR()	count
1	1	1	1
2	2	3	2
3	4	5	3
4	6	0	3

```
  10 REM PROGRAM II
  20 PRINT "This is all th
e program does"
  30 INPUT "Do you want an
other go? Y/N "letter$
  40 IF LEN(letter$)>1 OR
LEN(letter$)=0 THEN GOTO 30

  50 IF INSTR("YyNn",lette
r$)=0 THEN GOTO 30
  60 IF INSTR("Yy",letter$
)<>0 THEN GOTO 20
  70 PRINT "OK, I'll stop"
```

Program II

```
  10 REM PROGRAM III
  20 PRINT "This is all th
e program does"
  30 INPUT "Do you want an
other go? Y/N "letter$
  40 IF LEN(letter$)>1 OR
LEN(letter$)=0 THEN GOTO 30

  50 IF INSTR("YyNn",lette
r$)=0 THEN GOTO 30 ELSE IF
INSTR("Nn",letter$)<>0 THEN
 PRINT "OK, I'll stop" ELSE
 GOTO 20
```

Program III

```
   10 REM PROGRAM IV
   12 INPUT "Give word to b
e searched " search$
   14 INPUT "Give target to
 be looked for " target$
   16 count=0
   18 length=LEN(search$)
   20 FOR start=1 TO length
   30 IF INSTR(search$,targ
et$,start)<>0 THEN count=co
unt+1
   40 NEXT start
   50 PRINT "There are ";co
unt" occurrences of "target
$" in "search$
```

Program IV

tempting. It seems just made for a FOR ... NEXT loop, doesn't it?

Couldn't one be used to, say, count the total number of matches between a target string and a search string? In other words, could we use a FOR ... NEXT loop to check every occurrence of e in *element?*

Program IV looks a likely method. It asks the user for a search string and a target string. Then it uses an INSTR inside a FOR ... NEXT loop with *start* as the control variable to work its way through the string. A variable *count* keeps track of the matches.

It's a nice, well structured program that, if you've typed it in properly, will run and give you a result. The trouble is, it's the wrong result!

Try finding the number of e's in the word elements and you'll find that it tells you that there are five! Can you figure out what's happened?

The problem lies with the fact that as the loop cycles we start the search for a match at each successive letter of the search string.

When we start at the first letter, it's an e so there's a match and *count* goes to 1.

The next time round we start at the second letter, l, and start searching for a match. Now the third character provides an e so *count* is increased to 2.

The next cycle has the search beginning at 3. This is an e so *count* goes to 3. And that's where the problem lies. We've counted that second e twice. Figure I shows the values of *start:*

```
INSTR(search$, target$,
start)
```

and *count* for each cycle of the loop.

The moral is; beware of using INSTR in FOR ... NEXT loops. However, good advice though that may be, it still leaves us with the problem of counting the number of matches between a target string and a search string. Program V shows how it's done.

Here a REPEAT ... UNTIL loop is used. This cycles until

```
   10 REM PROGRAM V
   20 INPUT "Give word to b
e searched " search$
   30 INPUT "Give target to
 be looked for " target$
   40 count=0
   50 length=LEN(search$)
   60 position=1
   70 REPEAT
   80 position=INSTR(search
$,target$,position)
   90 IF position<>0 THEN c
ount=count+1:position=posit
ion+1
  100 UNTIL position=0 OR p
osition>length
  110 PRINT "There are ";co
unt" occurrences of "target
$" in "search$
```

Program V

there are no more matches left or the INSTR would be trying to search from a non-existent character.

The heart of the program is formed by lines 80 and 90. The first checks for a match and if there is one, records its character position in *position.* If there's no match then *position* becomes 0.

Line 90 comes into play when a match has occurred, that is, *position* is non-zero. When this is the case, *count* is increased by one. Also *position* is increased by one.

This means that next time

```
   10 REM PROGRAM VI
   20 INPUT "Enter your dat
a string ", data$
   30 position=1
   40 occurs=0
   50 length=LEN(data$)
   60 REPEAT
   70 position=INSTR(data$,
"*",position)
   80 IF position<>0 THEN o
ccurs=occurs+1:PRINT "Occur
rence ";occurs" is at posit
ion ";position:position=pos
ition+1
   90 IF position=0 AND occ
urs=0 THEN PRINT "There are
 no asterisks in " data$
  100 UNTIL position=0 OR p
osition>length
```

Program VI

round the loop the INSTR search will start *after* the last matching string. Hence there's no double counting.

Figure II shows the values of *position* and *INSTR (search$, target$, position)* for each cycle of the loop.

Once you've understood that, have a look at Program VI, which uses the same technique to look at a string and tell you where the asterisks occur, if anywhere. We'll come to why we put asterisks in a string in a moment.

If you've followed the article so far you should have no problems seeing how it locates the positions of the asterisks and prints the messages.

The main problem is why we should want to do it in the first place! And to understand that, take a look at Program VII, last month's Program VIII.

If you cast your mind back you'll remember how we used this program to select information from three data strings. The strings contained information about three animals, the data being held in a strict format. Figures III and IV give the details.

You'll recall that although the program was powerful, it wasn't very flexible. The fixed fields made for problems if Eric

```
   10 REM PROGRAM VII
   15 REM LAST MONTH'S PROG
RAM VIII
   20 REM ****************
   30 DIM record$(3)
   40 FOR loop=1 TO 3
   50 READ record$(loop)
   60 NEXT loop
   70 REM ****************
   80 INPUT "Do you want na
mes? Y/N " test$
   90 IF test$="Y" OR test$
="y" THEN name=TRUE ELSE na
me=FALSE
  100 INPUT "Do you want sp
ecies? Y/N " test$
  110 IF test$="Y" OR test$
="y" THEN species=TRUE ELSE
 species=FALSE
  120 INPUT "Do you want ag
e? Y/N " test$
  130 IF test$="Y" OR test$
="y" THEN age=TRUE  ELSE ag
e=FALSE
  140 INPUT "Do you want we
ight? Y/N " test$
  150 IF test$="Y" OR test$
="y" THEN weight=TRUE ELSE
weight=FALSE
  160 PRINT
  170 REM ****************
  180 FOR loop=1 TO 3
  190 IF name THEN PRINT MI
D$(record$(loop),1,4)"  ";
  200 IF species THEN PRINT
 MID$(record$(loop),6,3)"
";
  210 IF age THEN PRINT MID
$(record$(loop),10,1)"  ";
  220 IF weight THEN PRINT
MID$(record$(loop),12,1)"
";
  230 PRINT:PRINT
  240 NEXT loop
  250 REM ****************
  260 DATA " TOM CAT 3 7"
  270 DATA " IAN RAT 2 1"
  280 DATA "ERIC BAT 1 1"
```

Program VII

was an aardvark or Tom a tarantula. Also the fact that the data in each field is padded out by spaces seems a bit wasteful.

And this is where the asterisks come in. How about,

→

```
  10 REM PROGRAM VIII
  20 REM ****************
  30 DIM record$(3)
  40 DIM asterisk(3)
  50 FOR loop=1 TO 3
  60 READ record$(loop)
  70 NEXT loop
  80 REM ****************
  90 INPUT "Do you want na
mes? Y/N " test$
 100 IF test$="Y" OR test$
="y" THEN name=TRUE ELSE na
me=FALSE
 110 INPUT "Do you want sp
ecies? Y/N " test$
 120 IF test$="Y" OR test$
="y" THEN species=TRUE ELSE
 species=FALSE
 130 INPUT "Do you want ag
e? Y/N " test$
 140 IF test$="Y" OR test$
="y" THEN age=TRUE  ELSE ag
e=FALSE
 150 INPUT "Do you want we
ight? Y/N " test$
 160 IF test$="Y" OR test$
="y" THEN weight=TRUE ELSE
weight=FALSE
 170 PRINT
 180 REM ****************
 190 FOR loop=1 TO 3
 200 position=1
 210 occurs=0
 220 length=LEN(record$(lo
op))
 230 REPEAT
 240 position=INSTR(record
$(loop),"*",position)
 250 IF position<>0 THEN o
ccurs=occurs+1:asterisk(occ
urs)=position:position=posi
tion+1
 260 UNTIL position=0 OR p
osition>length
 270 IF name THEN PRINT MI
D$(record$(loop),1,asterisk
(1)-1)
 280 IF species THEN PRINT
 MID$(record$(loop),asteris
k(1)+1,asterisk(2)-asterisk
(1)-1)
 290 IF age THEN PRINT MID
$(record$(loop),asterisk(2)
+1,asterisk(3)-asterisk(2)-
1)
 300 IF weight THEN PRINT
MID$(record$(loop),asterisk
(3)+1,length-asterisk(3))
 310 PRINT:PRINT
 320 NEXT loop
 330 REM ****************
 340 DATA "TOM*CAT*3*7"
 350 DATA "IAN*RAT*2*1"
 360 DATA "ERIC*BAT*1*1"
```

Program VIII

NAME	SPECIES	AGE	WEIGHT
TOM	CAT	3	7
IAN	RAT	2	1
ERIC	BAT	1	1

Figure III: Record data

Character position	1	2	3	4	5	6	7	8	9	10	11	12
Data	E	T I R	O A I	M N C		C R B	A A A	T T T		3 2 1		7 1 1

Figure IV: Record format

From Page 27

instead of having fields of fixed length in a field as in:

```
260 DATA " TOM CAT 3 7"
270 DATA " IAN RAT 2 1"
280 DATA "ERIC BAT 1 1"
```

using asterisks to separate the fields as in:

```
260 DATA "TOM*CAT*3*7"
270 DATA "IAN*RAT*2*1"
280 DATA "ERIC*BAT*1*1"
```

The data is still the same, only it's arranged differently, with asterisks used as markers between the fields. And as the strings in the second method are shorter, they use up less memory space.

In the first case we know that the name field is the first four letters of the data string. In the second case we know that the name field is everything up to the first asterisk.

The species field lies between 6 and 8 for the fixed field method while in the asterisk method it lies between the first and second asterisks. In Program VII we can extract information from the fixed fields because we know where they start and finish.

With our asterisk method of storing data we don't know exactly where the data fields begin and end but we do know where they are in relation to the asterisks.

To use this second method of storing data all we need is a way of finding out where the asterisks occur in a string. And that's exactly what we did in Program VI.

So, by combining the methods of Program VI and VII we can use the second method of storing data, where the field lengths aren't fixed but vary. Program VIII is the result.

Although it looks fairly complicated, if you've followed Program VI and Program VII it should hold no terrors for you. All it is is Program VII adapted to use the asterisk method of marking fields in data strings.

Most of the lines of Program VIII are the same as those of Program VIII. Where they aren't, it's because some lines from Program VI have been introduced to deal with the new format of the data strings.

You'll see that lines 200 to 260 are almost identical to lines 30 to 100 of Program VI.

So really you've done it all before. The only problem may be with all the occurrences of *asterisk( )* in the MID$ of lines 270 to 300.

Although there's a lot of them, it's just simple maths, figuring out the length of each field from the positions of each asterisk.

Once you've convinced yourself that you understand the program, just see how flexible it is.

Not only does using the asterisks allow you to save memory space, but it also allows us to solve the aardvark problem.

Try changing the DATA lines to:

```
340 DATA ERIC*AARDVARK*12*
23
350 DATA TOM*TARANTULA*34*
3456
360 DATA HILDA*COW*666*666
```

and you'll find that the program handles it with ease. Try doing that the old way with Program VII.

So, this asterisk method – properly called the variable field method – is obviously a lot more flexible than our old fixed fields. Also it saves memory space.

But, as in everything, there's a drawback. In this case it's the fact that the program is a lot more complicated and slower than with fixed fields.

This is because it has to figure out where they begin and end, whereas with fixed fields it already knows.

So when you use strings to store information you've got two choices: fixed length fields or variable length fields. Which method you use depends on what you want.

And that's where we leave Eric, Tom and friends and also where we leave string handling.

Next time we'll be looking at some of the Electron's standard functions, powerful tools that will increase the scope of your programming enormously.

A COMBAT READY PHANTOM II WILL COST YOU £14M

"Well produced and technically good futuristic combat flight simulation featuring some of the best and smoothest 3D graphics I've come across on the Beeb." –
Computer Trade Weekly

RATED 8 OUT OF 10 FOR QUALITY, GRAPHICS, PLAYABILITY & VALUE

More than 'just' a very fast full-flight Simulator 'Phantom Combat' offers the excitement and energy of 1500 mph air to air combat in high resolution 3D colour graphics. This 100% machine code package has been written by a military flight simulator software engineer together with the B.A. captain who wrote the best selling '747' simulator for Doctor Soft

A SIMULATOR INSIDE A SIMULATOR

In one of the training modes (formation mode, FORM) it is **actually possible to fly the Phanton AND control the Enemy aircraft** which can be clearly seen flying in 3D outside your fully equipped cockpit. Alternatively, a friend can pilot the 'joint ion' (separate keys) while you attack.

THE ADVERSARY

Now, at last, enemy aircraft are NOT shown as arcade 'sprites', **they are computer drawn, navigated and 'flown' at a smooth 15 Frames per second**. The delta outlines reflect Soviet Mig 21 (Fishbed)/9Su 15 (Flagon) performance. In combat mode (CBAT) they fight back, intelligent and dangerous.

INSIDE

Instrumentation is comprehensive with a wealth of clear and precise displays, **featuring both analogue and digital readouts**, eg speed in knots as shown on a dial AND digitally, with a separate Mach number display, radar computed target range altitude and bearing shown, target pointer and gunsight; military 'Tacan' navigation (Tactical air navigation) etc.

'OUTSIDE'

External views includes Horizon, other aircraft, a network of ground detail points, **separate landing runways and animated 'strobe' approach lighting**. NO 'chunky' pixels, all objects are drawn in fine, high resolution coloured lines. **The view is recomputed and redrawn 15 times every second**

AIR TO AIR COMBAT

Warning There are no 'lives' but your single one, your score is zeroed and the program restarted if you are shot down or crash. This motivates you to try and 'bring home' a damaged aircraft. May different forms of damage can occur. Most are survivable, eg a gear up runway landing if smooth enough. **If you can't land, use the EJECTION seat and survive**

THE BEST ALTERNATIVE?.. £9.95

PHANTOM COMBAT
Flight Simulator
3rd generation flight software
by Nick Brown and Roger Selby

BBC 32K
ELECTRON versions

DOCTOR SOFT
simulations
Doctor Soft Advanced Software

P.O. Box 66 · East Preston · Littlehampton
West Sussex · BN14 2TX

Telephone: (0903) 770044
Accounts: (0903) 776000
Telex: DocSoft 878152 (WCTC)

Available at your Local Dealer, distributed by Microdealer, Leisuresoft, & Centresoft. Also direct from Doctor Soft:

MAIL ORDER
FIRST CLASS BY RETURN,
POST FREE

BBC ☐ ELECTRON ☐ BBC DISK* ☐

*Add £3 for disk version. Cheques/PO payable to "Doctor Soft Ltd" or state Access No. Overseas: add Airmail at cost.

Name .....

Address .....

.....

.....

How to catch a sprite on the hop

SO far in this short series we've looked at how to print multi-coloured sprites, read the keyboard and move the sprites around the screen in response.

This month we're going to look at simple collision detection.

I'll also show you how to keep track of where the sprites are and keep them within any preset boundaries.

Collision detection is fairly easy if you go about it in the right way – and peeking the screen memory is definitely the wrong way.

To keep track of where sprites are on the screen, or off, it's best to store their position as a pair of x, y coordinates.

The sprites' coordinates aren't the same as the normal coordinates that we PLOT at, though the screen is 40 columns wide and 256 rows high, and these values are more convenient to use. The top left corner of the screen has coordinates 0,0.

When it comes to printing a sprite there is a problem. The coordinates aren't much use for the print routine as it needs a screen address.

There are two ways round this. The hard way would be to calculate the screen address from the coordinates.

I've never been much of a mathematician, so my method is simply to keep a record of both the screen address and coordinates of each sprite. We saw how to read the keyboard in our September, 1985, issue and looked at the print routine in the November issue.

If you have a look at this month's listing you'll see that it is a combination of the last two programs, so most of it should look familiar.

If you run it you'll see that two sprites are placed on the screen. One can be moved by pressing the cursor keys and the other is fixed.

Try moving the sprite around the screen and see what happens. There are two important points to notice:

- The sprite cannot be moved out of the box, left, right, top or bottom.
- Whenever the two sprites collide the Electron will beep.

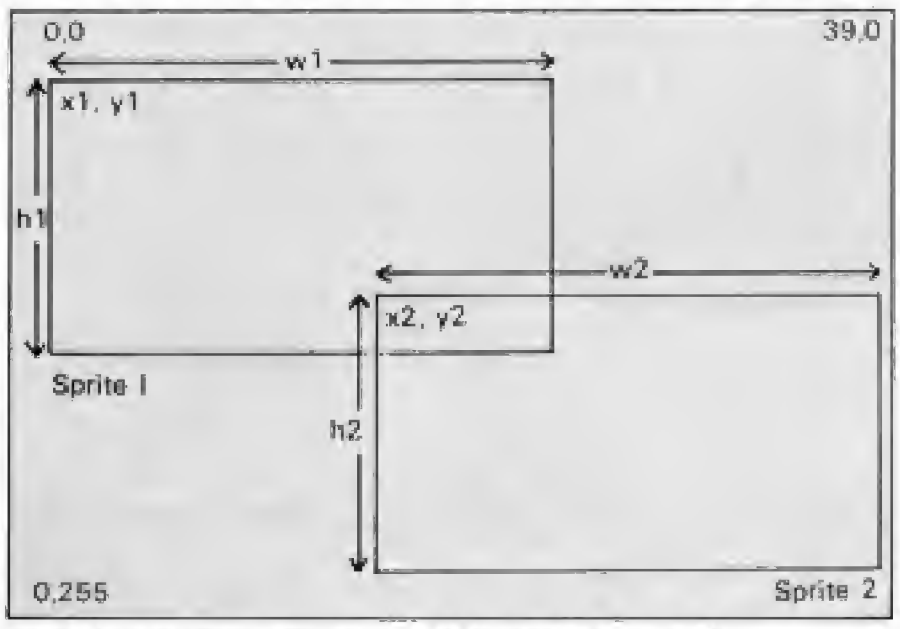


Figure 1: Detecting collisions between sprites in Mode 5

The coordinates of the fixed sprite are stored in *ax%* and *ay%*, and the coordinates of the moving sprite in *x%* and *y%*.

The initialisation part of the program starting at line 300 sets the start coordinates and addresses and places the sprites on the screen.

In order to keep track of the screen coordinates the routines *up*, *down*, *left* and *right* have been modified.

When the sprite is to be moved the relevant x or y coordinate is checked to see if it is at the edge of the box.

Take a look at the *up* routine, starting at line 910. After checking that the cursor up key is pressed the y coordinate is compared with 80.

If it is equal to 80 then the sprite must be at the top of the box so it can't be moved any higher.

A branch is then made to *notup* which simply returns from the subroutine.

If *y%* is not equal to 80 then 4 is subtracted and the sprite is moved up four pixels.

Remember, each of the lines we PRINT on is eight bytes deep in the screen RAM corresponding to eight vertical pixels.

As we're moving the sprite up and down four pixels at a time it can be either exactly on a line or half way down it.

If you examine the three least significant bits of the address you'll find that they are either 000 at the top of a line or 100 half way down.

To find out where it is the least significant byte is ANDed with 7. If it's zero it's at the top.

Each line is &140 bytes long, so subtracting &140 would move the sprite to the top of the line above.

This means that &13C (&140 - 4) must be subtracted from the address to move it from the top of one line to half way down the one above.

If it's currently half way up a line then all that is neccessary is to set the three least significant bits of the address to zero. This is achieved by ANDing the least significant byte with &F8.

The other sections – *down*, *left* and *right* – work in exactly the same way, check for the key then check the coordinates to see whether it can be moved.

Now that we have the coordinates of each sprite we can easily test for collisions.

The actual collision detec-

Part 5 of ROLAND WADDILOVE's series on programming graphics with arcade games in mind.

tion is carried out by *collision* starting at line 610 which returns with the Carry flag set if collision has occurred.

The routine needs the coordinates and the sizes of the two sprites and *bumped* sorts this out.

The coordinates are placed in *x1, y1, w1, h1* and *x2, y2, w2, h2*, an eight byte block of workspace.

A point to note is that one must be subtracted from the heights and widths so a collision is detected when they *overlap*, and not when they are simply next to each other.

The Econet workspace from &90 to &9F is used by *collision*. Since Econet isn't available for the Electron this area of memory is free.

To test whether two sprites are overlapping the x coordinates are first tested then the y coordinates. Here's the algorithm used, where 1 refers to sprite 1, 2 to sprite 2 and h and w are the heights and widths:

IF x1 is less than x2 THEN add w1 to x1, see if this is greater than x2 and RETURN if false ELSE add w2 to x2, see if this is greater than x1 and RETURN if false.

IF y1 is less than y2 THEN add h1 to y1, see if this is greater than y2 and RETURN ELSE add h2 to y2, see if this is greater than y1 and RETURN.

If you're not sure how this works Figure I shows two overlapping sprites. In this case x1 is less than x2 so w1 is added to x1. This is greater than x2, so sprite 1 must be overlapping sprite 2's left hand edge.

The y coordinates are now checked, as sprite 1 could be at the top of the screen and sprite 2 at the bottom.

In this case y1 is less than y2, so h1 is added to y1. This is greater than y2 so sprite 1 has definitely been in collision with sprite 2.

Study *collision* to see how this is carried out in machine code.

There are a couple of other important routines in this month's listing. The first is an Osword call to produce a sound when the sprites collide.

The sound data is at line 590. EQUD is used to store the data four bytes at a time in an eight byte parameter block. Figure II describes the layout.

Byte	Parameter
0	Channel.
1	Should be zero.
2/3	Volume/envelope.
4	Pitch.
5	Should be zero.
6	Duration.
7	Should be zero.

Figure II: Parameter block used by Osword with A=7

The parameters passed are the same as those used by the Basic SOUND command. The only difference is that a few zero bytes are needed as padding, and the volume should be in two's complement arithmetic.

Since any negative number produces maximum volume on the Electron &FFFF (−1) would be satisfactory.

To produce the sound the X and Y registers are loaded with the high and low bytes of the address of the block, A with 7 and a call to Osword at &FFF1 is made. Line 550 shows the technique.

The second interesting routine is my own equivalent of *FX19. We've seen that this drastically reduces flicker.

However, like all operating system calls, it's slow. The following routine does the same job only much quicker:

```
SEI
LDA #4
.frame
BIT &FE00
BEQ frame
```

This can be seen near the start of the print routine at line 1170.

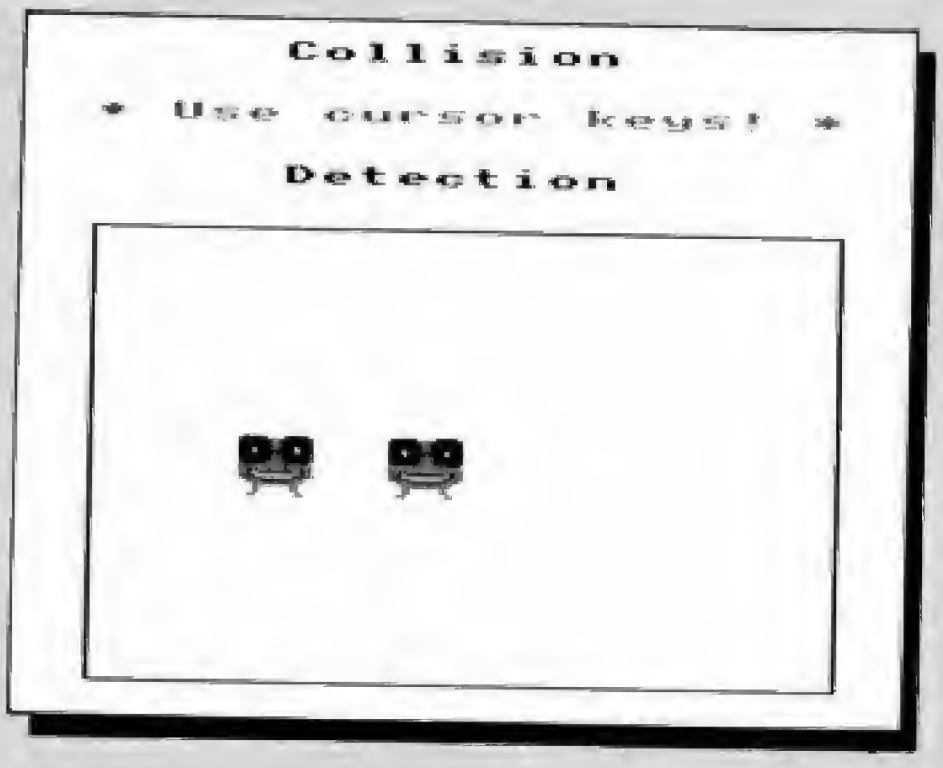
Bit 3 of &FE00, one of the registers in the ULA, is set when frame flyback occurs. It's much quicker to test the bit itself than go through the legal Osbyte call.

That's all for this month. By now you should now have some powerful machine code routines for writing arcade games. All you need is an original idea.

```
  10REM Collision Detectio
n
  20REM By R.A.Waddilove
  30REM (c) Electron User
  40ON ERROR GOTO 60
  50*FX163,128,1
  60ON ERROR OFF
  70FOR byte=0 TO 95
  80READ data
  90byte?&C00=data
 100NEXT
 110PROCassemble
 120MODE 5:VDU 23,1,0;0;0;
0;
 130DRAW 0,700:DRAW 1278,7
00:DRAW 1278,0:DRAW 0,0
 140COLOUR 2:PRINT'SPC(5)"
Collision"
 150COLOUR 1:PRINT''"* Use
 cursor keys! *"
 160COLOUR 2:PRINT'SPC(5)"
Detection"
 170CALL &900
 180END
 190
 200DEF PROCassemble
 210old=&70:new=&72:rows=&
74:columns=&75:tempcol=&76:
temp1=&78
 220address=&80:xX=&82:yX=
&83
 230alien=&84:axX=&86:ayX=
&87
 240x1=&90:y1=&91:w1=&92:h
1=&93:x2=&94:y2=&95:w2=&96:
h2=&97
 250osbyte=!&20A AND &FFFF
:osword=!&20C AND &FFFF
 260FOR pass=0 TO 2 STEP 2
 270P%=&900
 280[ OPT pass
 290
 300.initialise
 310LDA #16:STA axX:LDA #1
60:STA ayX  \set alien x,y
 320LDA #&80:STA alien:STA
 new:LDA #&71:STA alien+1:S
TA new+1
 330LDA #&00:STA newdata+1
:LDA #&0C:STA newdata+2
 340LDX #4:STX columns:LDY
 #24:STY rows:LDY #0:JSR pu
t
 350LDA #8:STA xX:LDA #160
:STA yX  \set your x,y
 360LDA #&40:STA address:S
TA new:LDA #&71:STA address
+1:STA new+1
 370LDA #&00:STA newdata+1
:LDA #&0C:STA newdata+2
 380LDX #4:STX columns:LDY
 #24:STY rows:LDY #0:JSR pu
t
 390
 400.start
 410JSR bumped
 420LDA #&00:STA olddata+1
:STA newdata+1:LDA #&0C:STA
 olddata+2:STA newdata+2
 430LDA address:STA old:LD
A address+1:STA old+1
 440JSR readkeys:BEQ escap
e
 450LDX #4:LDY #24:JSR pri
nt
 460.escape  \INKEY(-113)
 470LDA #&81:LDX #&0F:LDY
#&FF:JSR osbyte:TYA:BEQ sta
rt  \key pressed?
 480RTS  \return to Basic
 490
 500.bumped  \set up block
 for collision detection
 510LDA xX:STA x1:LDA yX:S
TA y1:LDA #3:STA w1:LDA #23
:STA h1
 520LDA axX:STA x2:LDA ayX
:STA y2:LDA #3:STA w2:LDA #
23:STA h2
 530JSR collision
 540BCC nothit
 550LDX #sound MOD256:LDY
#sound DIV256:LDA #7:JSR os
word
```

From Page 31

```
  560.nothit
  570RTS
  580
  590.sound EQUD &FFF80011:
EQUD &00010060
  600
  610.collision
  620LDA x1:CMP x2:BCC c1
\x1>x2 ?
  630LDA x2:CLC:ADC w2:CMP
x1  \x2+w2>x1 ?
  640BCS checky:RTS
  650.c1 ADC w1:CMP x2  \x1
+w1>x2 ?
  6600CS checky:RTS
  670.checky
  680LDA y1:CMP y2:BCC c2
\y1>y2 ?
  690LDA y2:CLC:ADC h2:CMP
y1  \y2+h2>y1 ?
  700RTS
  710.c2 ADC h1:CMP y2  \y1
+h1>y2 ?
  720RTS
  730
  740.readkeys
  750\right  INKEY(-122)
  760LDA #&81:LDX #&86:LDY
#&FF:JSR osbyte:TYA:BEQ lef
t  \key pressed?
  770LDA xX:CMP #36:BEQ not
right:INC xX
  780CLC:LDA address:ADC #8
:STA address:STA new  \addr
ess=address+8
  790LDA address+1:ADC #0:S
TA address+1:STA new+1
  800.notright
  810RTS
  820
  830.left  \INKEY(-26)
  840LDA #&81:LDX #&E6:LDY
#&FF:JSR osbyte:TYA:BEQ up
 \key pressed
  850LDA xX:BEQ notleft:DEC
 xX
  860SEC:LDA address:SBC #8
:STA address:STA new  \addr
ess=address-8
  870LDA address+1:SBC #0:S
TA address+1:STA new+1
  880.notleft
  890RTS
  900
  910.up  \INKEY(-59)
  920LDA #&81:LDX #&C6:LDY
#&FF:JSR osbyte:TYA:BEQ dow
n  \key pressed?
  930LDA yX:CMP #80:BEQ not
up:SEC:SBC #4:STA yX
  940LDA address:AND #7:BEQ
 up1
  950LDA address:AND #&F8:S
TA address:STA new:LDA addr
ess+1:STA new+1:RTS
  960.up1
  970SEC:LDA address:SBC #&
3C:STA address:STA new  \ad
dress=address-&13C
  980LDA address+1:SBC #&1:
STA address+1:STA new+1
  990.notup
 1000RTS
 1010
 1020.down  \INKEY(-42)
 1030LDA #&81:LDX #&D6:LDY
#&FF:JSR osbyte:TYA:BEQ not
down
 1040LDA yX:CMP #232:BEQ no
tdown:CLC:ADC #4:STA yX
 1050LDA address:AND #7:BNE
 down1
 1060LDA address:CLC:ADC #4
:STA address:STA new:LDA ad
dress+1:STA new+1:RTS
 1070.down1
 1080CLC:LDA address:ADC #&
3C:STA address:STA new  \ad
dress=address+&13C
 1090LDA address+1::ADC #&1
:STA address+1:STA new+1
 1100.notdown
 1110RTS
 1120
 1130.print  \uses new/old/
X=columns/Y=rows/olddata/ne
wdata
 1140STX columns:STY rows
 1150STX tempcol  \save col
umns
 1160LDY #0
 1170SEI:LDA #4:.frame BIT
&FE00:BEQ frame  \#FX19
 1180.loop1
 1190LDA old:STA temp1:LDA
old+1:STA temp1+1  \save ad
dress of column
 1200LDX rows
 1210.loop2
 1220.olddata LDA &3000:EOR
(old),Y:STA (old),Y
 1230INC olddata+1:BNE p1:I
NC olddata+2
 1240.p1 LDA old:AND #7:CMP
#7:BEQ bottom
 1250INC old:BNE p2:INC old
+1:.p2 BNE next1
 1260.bottom  \row
 1270CLC:LDA old:ADC #&39:S
TA old:LDA old+1:ADC #1:STA
 old+1
 1280.next1
 1290DEX:BNE loop2  \next r
ow
 1300LDA temp1:ADC #8:STA o
ld:LDA temp1+1:ADC #0:STA o
ld+1
 1310DEC columns:BNE loop1
 \next column
 1320LDA tempcol:STA column
s  \restore columns
 1330.put  \put sprite on s
creen
 1340.loop1
 1350LDA new+1:STA temp1+1:
LDA new:STA temp1  \save ad
dress of column
 1360LDX rows
 1370.loop2
 1380.newdata LDA &3000,Y:E
OR (new),Y:STA (new),Y
 1390INC newdata+1:BNE p3:I
NC newdata+2
 1400.p3 LDA new:AND #7:CMP
 #7:BEQ bottom2
 1410INC new:BNE p4:INC new
+1:.p4 BNE next2
 1420.bottom2  \row
 1430CLC:LDA new:ADC #&39:S
TA new:LDA new+1:ADC #1:STA
 new+1
 1440.next2
 1450DEX:BNE loop2  \next r
ow
 1460LDA temp1:ADC #8:STA n
ew:LDA temp1+1:ADC #0:STA n
ew+1
 1470DEC columns:BNE loop1
 \next column
 1480CLI:RTS
 1490]
 1500NEXT
 1510ENDPROC
 1520
 1530REM Sprite Data
 1540REM rows=24/columns=4
 1550DATA 112,247,247,247,2
47,230,230
 1560DATA 247,247,247,120,1
5,15,60,44
 1570DATA 30,15,7,3,1,1,1,3
,3,192,237
 1580DATA 237,237,237,252,2
52,237,237
 1590DATA 237,195,15,15,15,
240,0,135
 1600DATA 120,15,15,0,0,0,0
,48,123,123
 1610DATA 123,123,243,243,1
23,123,123
 1620DATA 60,15,15,15,240,0
,30,225,15
 1630DATA 15,1,1,0,0,224,25
4,254,254
 1640DATA 254,118,118,254,2
54,254,225
 1650DATA 15,15,195,67,135,
15,14,12,0
 1660DATA 0,0,12,12
```



This listing is included in this month's cassette tape offer. See order form on Page 61.

MORE HINTS TO SEND YOU ON YOUR WAY...

ONCE again my postbag has been bulging with the wisdom gained by other adventure addicts.

Mystery Fun House seems to be creating quite a lot of problems for Richard Newton and Mark Constantin.

Mark writes that he cannot seem to get started at all, and is just wandering around the first few locations. To get started, have a look at the tree and the grating.

The heel of your shoe holds something useful, and a good chew will stick a coin in the bank – if you have the right branch.

The valve turns the merry-go-round music off and a blue button answers another problem.

Climbing the pole that holds the horse upright will lead you partway to the catwalk.

J. Hughes, A. Lovell, G. Ross and Matthew Gaunt are stuck in ***Twin Kingdom Valley***. There are two routes to the giant and the princess, from the clearing and down, in the castle.

For the first you will need the gold and bronze keys. From the clearing go D,N,N,W, W,D,D,N,N,N,N,N,N.

From the grand stairs in the castle go D,D,E,E,E,D,D,D,N.

To stop the princess fleeing you must be wearing the amulet. When you have done everything, have a second look at the treasures you have found.

Wheel of Fortune has also raised a lot of questions this month. D.A. Calvert, Alan Riddell and A. Lovell are all stuck in it.

To open the trapdoor you must unbolt it. You can't get past the farmer until after you have gone through the trap door.

You cross the canal from the other side. You will find something to go into the basket that the troll doesn't like.

To kill the fly and get the wheel you must lure the fly into the spider's cave. You can't keep the trapdoor open.

The beggar goes home for his tea after he has lowered you down the well. To get him to do this you must give him the penny and ask him nicely.

But make sure you get something with a penny in and beat him to the machine.

The well leads to the other locations in the game, not the trapdoor.

Keith Inman (that's two mentions Neil), Mark Rummings, Jason Owens and Andrew Stewart need help with ***Greedy Dwarf.***

Jump over the trip wire or you will be stuck in the maze.

Map out the maze by dropping objects if the locations all look the same. Be fast on your feet to get past the shadow monsters. The hut is across the river.

Wave the wand to cross the chasm. You need the compass to get out of the muddy maze.

Alan Riddell wants to know the following about ***Hampstead: Where is Pippa?*** At the party near the bank. *Is there anything in the industrial estate?* Yes! *How can you get the money for the cottage?* Vote option 3 in the board-room. Wear the tweed suit and have the object from the industrial estate to meet the man on the train.

Alan is also stuck in ***Adventure*** along with Peter Young. To get into the cavern at the start say Open Sesame.

Ian Bevan has sent in a map for ***Sorcerer of Claymorgue Castle.*** He has found all 13 stars, but doesn't know where he has to store them. Can anyone help?

He has enclosed a solution to his travels so far and anyone who wants a copy should write in enclosing an SAE.

Mark Allen writes asking for more tips on ***Twin Kingdom Valley***. See the July 1985 column for the special I did on it, Mark.

Iris Rose has sent in solutions to questions raised in previous months about ***Galadriel in Distress*** and ***Pettigrew's Diary.***

Annie Gramm is in the hotel. Wait until the fans are outside then go in and ask for Barry Manilow (anagram of Lim Ray Barnow).

Rhythma Tiq is in the house on the island. The combination lock code is 58. Help the Japanese tourists, they will give you a camera.

Take a picture of Annie and show it to Rhythma. The money is in Cleopatra's Needle. Say "kulfatyn" to the old man on the park bench.

In ***Galadriel,*** to cure the toad poisoning show the bottle and the goblet to the lore-master.

Neil Sedgwick has taken me to task for spelling his name wrong, twice. Sorry Neil! He also asks me to mention his friend Keith Inman. Your wish is my command!

Yes, Chris Gregson, the anagram in ***Sphinx*** is "kneel, wave wand".

C. Nixon wants to know why the rabbit is described as being "a giant" in ***Sphinx.*** This is supposed to be a hint to let you know that it is heavy.

Iris Rose thinks the sound in ***Sim*** is terrible and wants to know if it can be turned off. Try *FX210,1 before you load the program.

John Tipper can't get the lumber and shovel through the crack, find the key to the locked door, or get the flotsam and jetsam in ***Pirate Adventure.***

You will find a key under the rug but you are on the wrong side of the door – look for another route to it. You can't get the flotsam and jetsam.

A. Lovell would like some answers to his problems in ***Kingdom of Klein.*** You can't get unfrozen in the icicle room, get out of the rowing boat in the pool, or kill the squid.

To get into the sandcastle after you have waved the wand you need a password. This is made up of the letters you find in the game.

In response to Paul Edman's pleas for help, here's a list of treasures in ***Classic Adventure*** – I know there are 16 in the list, and there should only be 15, so obviously one of them isn't a treasure: Emerald, platinum pyramid, vase, spices, gold chain, eggs, silver, jewellery, coins, pearl, pillow, rug, treasure from chest, gold nuggets, diamonds and the trident.

M. Frazer should drop the

From Page 33

rod if he wants to get the bird in this adventure.

M. Rodger has asked for help with the black maze in ***Castle of Riddles.*** There is only one route through the maze. Find it and go back for the bomb.

A password will find you in the shooting gallery and the bomb will frighten off the giants.

M. Frazer wants to know how to get through the musical door in ***Fantasia Diamond.*** Give the baton to the conductor and the violin to the violinist. Drop a book and tell them to play the music in it.

Paul Edmans and Richard Newton have problems with ***Blue Dragon.*** You can't get past the slug.

Swap the gold with the smug dwarf to get a gem that you can use to open the door with the outline. After you have freed the princess you must go and kill the dragon.

Andrew Stewart and Iris Rose are stuck in ***Sadim Castle.*** Read the Bible twice to avoid getting your throat cut. Use the stake on the portcullis. Kill the cheetah with the knife.

Alex Bateman keeps losing the tent stake and can't open the coffin and Matthew Captain can't find the coffin in ***The Count.***

"A century of dust" provides an answer to the stake.

The coffin is in the crypt – try smoking a cigarette. Try opening the coffin then go inside it.

Iris Rose wants to know how to kill the mad monk in ***Ring of Time.***

Use the sword past the crocodiles.

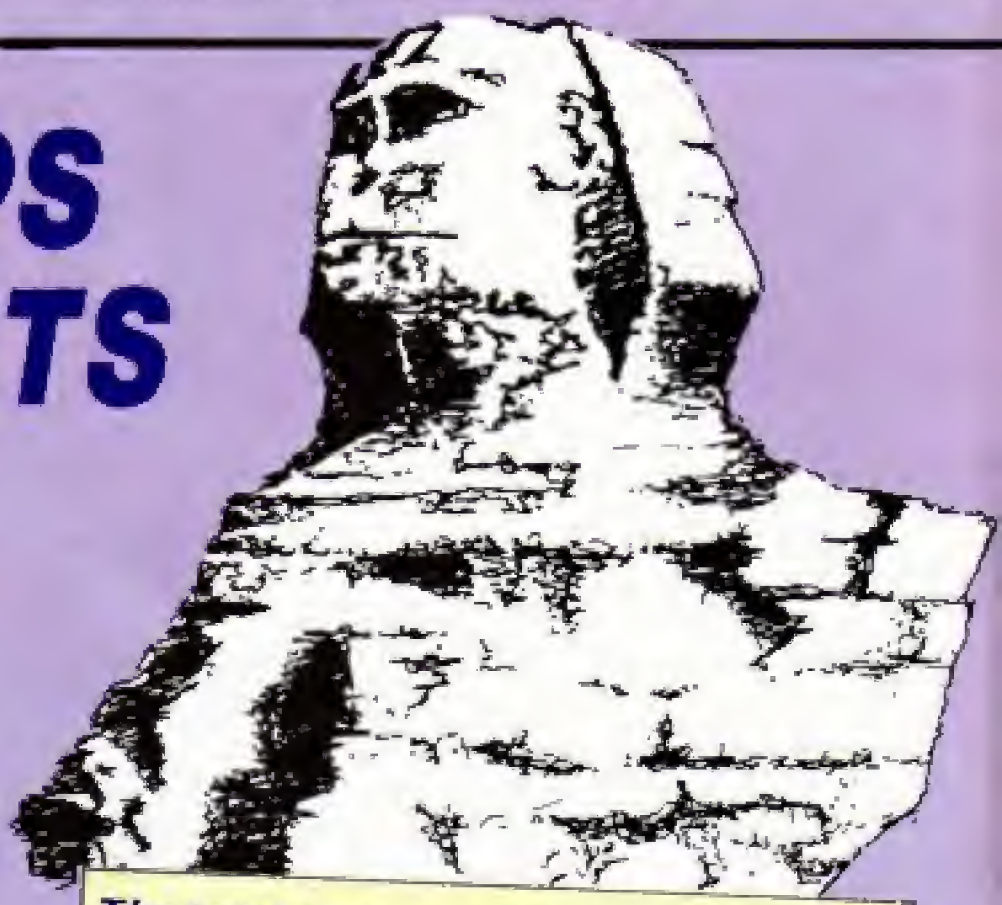
Alan Riddell wants to know how to get the key-box and sword in ***Stolen Lamp*** without the roof collapsing. You only need one of these items.

He also wants some help with ***Spiderman.*** Use something to stop the fan from turning. Mix the chemicals in the lab to make some web fluid.

Finally, Iris Rose and Alice Morland are stuck in ***Crown Jewels.*** Turn on the torch in the chamber of horrors, but remember there are two of them.

Get the matches and set light to the highly inflammable waxwork.

SPHINX KEEPS SOME SECRETS



ANOTHER busy month, particularly when you're as committed as I am to solving adventures honestly, without help of hint sheet or devious strategems. And if you believe that . . .

I must confess that I had to cheat in order to finish ***Sphinx Adventure*** without getting killed by that blasted dwarf.

I altered a message in line 194 and deleted the GOTO 449 so that the dwarf couldn't kill me.

I also changed the variable *C* from −3 to −100 in line 77, allowing me to have an almost unlimited inventory.

Anyway, thanks to all that cheating, I can finally bring you the Sphinx special that I have promised.

A total of 630 points is scored by collecting all the treasures and a further 170 points, to make the total of 800, is given for dropping them at the Sphinx, kneeling and waving.

An interesting point is that I got 30 points for getting the boat, which I then lost when it was destroyed. I wonder if this was a result of altering the program? Anyway, here are a few pointers on general strategy:

Map everywhere and find the routes that avoid the pirate and the rabbit.

Go across the everglades, collect everything and return.

Go across the troll's bridge and, taking care to avoid the bear, collect the jack then rub the ring to get out.

Come back through the fiery passages, and collect the treasure in the safe.

Explore all the locations except the iron passages and coloured rooms maze.

Cross the glacier and get the stilton. Get as much treasure as possible and leave what is not needed at the Sphinx.

Go to the vampire's castle and get the mouse and boat.

Go back across the glacier and get the matches past the elephant.

Rub the ring then go up to the lake and cross it.

Get the crown and return to the sorcerer's lair.

Go back up to the Sphinx, making sure that you have collected any treasure left lying around.

Drop everything, kneel and wave the wand.

The treasures

Bottle *found on the road north of the starting point.*
Carrot *in the gardener's store. Don't give it to the rabbit!*
Bar of silver *in the smelting room past the fiery passages.*
Rug *in the oriental room.*
Library books *in the library.*
Set of coins *in the treasury over the swamp.*
Bar of gold *at the end of the yellow brick road.*
Cluster of opals *in the dead end, west of the music room.*
Mithril ring *in the fairy grotto.*
Bar of platinum *in the safe.*
Emerald *in the green room.*
Pearls *inside the clam.*
Cushion *in the soft room.*
Jar of spices *in the spice room.*
Diamond *in the diamond mine.*
Bracelet *in the rainbow room past the goblins.*
Sceptre *in the pirate's hideout.*
Rubies *in the pirate's chest.*
Cluster of sapphires *in the blue room.*
Amethyst *in the north tower of the vampire's castle.*
Crown *in the west ante-chamber across the lake.*

Help needed

Peter Laughton wants to know where ***Xanadu*** ends. He has got into the castle and the adventure that's being made but doesn't know what's next. Can anyone help him?

Charles Forrest has scored 97% in ***Hampstead***. If anybody has fully solved this can they help us both? If you want a copy of his solution so far send an SAE.

Alice Morland is having problems with one of Peter Gerrard's adventures, ***The Tunnel***. She wants to know how to get past the panther.

Iris Rose wants to know which barber's shop she should enter in part 3 of ***Pettigrews Diary*** and why.

NOW and then I get a really superb set of hints that deserve a section to themselves. R. Henderson earns this honour by providing some very useful hints for three popular and non-too-easy games. He must spend as much timme on adventures as I do!

VOODOO CASTLE

- *Saw your way out of the cell.*
- *Wave the ring to open the stone door.*
- *Mix and drink the chemicals to get through the small door.*
- *Drink the witches' brew for a moving experience!*
- *To get the ju-ju bag, get the statue and go to the room where you found the bag. Then say "zap" and listen.*
- *Listen to what the raven has to say.*
- *If you press Break by accident typing in CALL &1902 will allow you to restart.*

CROWN JEWELS

- *Don't drink the wine in the Queen's bedroom.*
- *Don't go where the PM is.*
- *To get the orb, burn the statue.*
- *The sceptre is behind Big Ben's clockface.*
- *The crown is under the throne in the palace.*

STRANDED TIPS

- *The picklock is in the desert.*
- *To make the ship take off, "drop fuel" and "pull lever".*
- *To start the Tardis a time crystal will come in handy. A suit is necessary attire here as visitors to the engine room will discover.*
- *After dropping the crystal, type press <colour> and go south.*
- *The house key is in the cave.*
- *Unlock your front door and you have done it!*

Magic

The wand: This is found in the sorcerer's lair. Wave it at the chasm and the glacier to create a bridge. In the fairy grotto wave it to get the mithril ring. When you have collected all the treasure and taken it to the sphinx you must kneel and wave it again.

The mithril ring: You'll find this in the fairy grotto. When rubbed it will transport you to the sorcerer's lair. Use it at the rockfall to return across the swamp and the lake. It's also useful for short cuts.

Diaxos: This is whispered to you in the cavern near the furniture room. Use it to open the safe and to get to the west antechamber from the spell chamber.

The lamp: Found in the smithy. This has a habit of running out. When it starts to get dim, rub it, although this does not work if the lamp actually goes out.

The characters

Elephant in the palace near the catacombs. Drop the mouse.
Goblins in the hall of the mountain king. Throw the dragon's teeth.
Pirate at the crossroads. He will steal things, usually the bottle, so map round him.
Rabbit in the straw room. Map round him and don't give him the carrot. If he follows you across the troll's bridge it will collapse.
Crocodile at the edge of the everglades. He's hungry, so feed him.
Troll at his bridge – where else? Pay him and you'll get rid of him for good.
Bear in his cave. He's so friendly that he will follow you around. His weight will collapse the troll's bridge.
Ogre in the furniture room. Use the sword.
Dwarves crop up everywhere. Use the axe to kill them, but make sure you get it back. Their movements are all randomised so they can kill you before you can even throw the axe. You'll have to keep trying, anyway.
Dragon in his lair. Use your bare hands then look around.
Orc in the passage after the soft room. If the bear is following you he will chase it away.
Mouse in the dungeon in the vampire's castle. Feed him.
Vampire in his castle. Open the casket in the courtyard and use the stake.

Additional objects

Food in the banqueting hall.
Keys in the smithy. Use them to open the pirate's chest.
Sword in the sword chamber. Use it to kill the ogre.
Water at the lake. Fill the bottle and throw the water to get past the fiery passages.
Stake in the canyon north of the yellow brick road (handy for killing vampires).
Jack at the bottom of the rockfall.
Dragon's teeth appear after you have killed the dragon.
Clam in the oyster room. Use the jack to open it.
Casket in the vampire's castle.
Boat just outside the castle.
Matches in the laboratory.
Stilton in the dairy past the goblins.
Chest in the pirate's hideout.
Sphinx somewhere in the desert.
Axe thrown by the first dwarf to appear. Make sure you get it then or you've no protection when the sword melts.

Other problems

Fiery passages: Fill the bottle at the lake, go to the passages and throw the water.
Catacombs: There are only four locations here, and there are three external exits to the palace, the plateau and the hall of the mountain king. Drop objects to make a map.
Glacier: You must have the ring when you try to cross the bridge.
Sphinx: Met in the desert. Go N,E,N,S,E,W,E,S,N,S,E,W,E,S,N,S,E,W,E,S,Nt
Serpent: To get out you must light the matches.
Lake: How to go across in the boat? Try CROSS LAKE.
Castle: To get to the vampire's castle go S,S,N,E,W,S,D,D,U from the bear's cave.
Storage: If you can't carry any more leave things at the Sphinx for safe keeping.
Capital letters: These are an anagram, KNEEL and WAVE WAND.

I have tried to incorporate the questions I get asked the most. You still have a lot to do though. For instance you don't just reverse the directions to get back from the sphinx and castle.

Never before have there been such money-saving offers for readers of a computer magazine!

EXPAND your Electron

. . . for much, much less than the price you'd normally p

How the Plus 1 helps you m the most of your Electron

With the Plus 1, you and your Electron enter a whole new computing dimension. The Plus 1 turns your Electron into a fully fledged micro capable of using printers, joysticks and cartridge ROMs – the software that comes on a chip. In addition, the Plus 1's analogue to digital port gives access to the outside world – while the slots for the ROM cartridges allow the Electron to take advantage of the latest, most exciting hardware developments yet to be released.

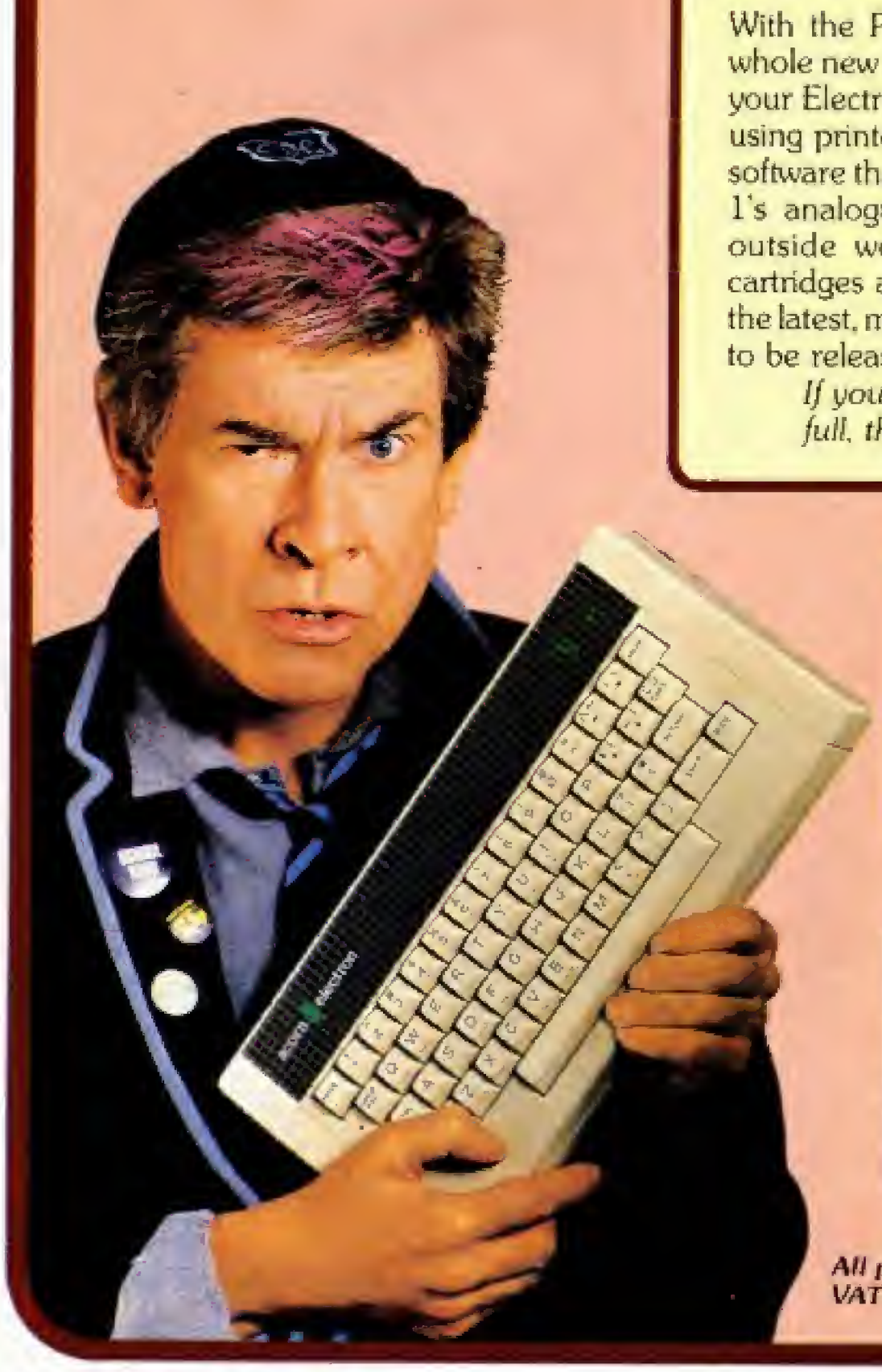
If you want to use your Electron to the full, then it's essential you get a Plus 1.

Normal

Spec for re Electi

£3

Inc. F ROM



EXPAND – with the Electron W

Now you can transform your Electron into a serious micr Workstation. This package consists of a Plus 1 and th wordprocessor and Viewsheet spreadsheet, both on car Workstation makes the Electron a hard-working yet inexpe and office. From business letters to your first novel, from l cash flow crises, it can take them all in its stride.

Normal price £119.80 *Electron User*

EXPAND – with the Electron Da

Bring the speed of discs and the organised memory of a po your Electron with the Electron Data Manager. It cons interface and Acorn's official disc-based Database – a disc keep track of things quickly, reliably and efficiently. Sophist fast, the Data Manager is a must for all serious Electron

Normal price £248.95 *Electron User*

All prices include VAT and carriage.

Please use the order form

ay!

nake

nce £59.90

al price
aders of
on User

9.95

EE game
cartridge

orkstation!

with the Electron
e acclaimed View
idge ROMs. The
sive tool for home
ome economics to

price £69.95

ta Manager!

werful database to
sts of the Plus 3
that allows you to
ted, accurate and
pplications.

price £149.95

on Page 61

EXPAND – with the Electron Language Lab

Now you can dramatically extend your programming horizons with this exclusive offer from Electron User. It comes complete with the Plus 1 interface and the Pascal and Logo cartridge ROMs. The Language Lab frees you from the limitations of Basic by giving you the two most educationally favoured high level programming languages – Logo and Pascal.

LOGO: *Acclaimed by teachers, parents and pupils alike, it's the language that's put the fun back into learning the fundamentals of programming. It's simple enough for a child to use, yet complex enough to satisfy the most exacting computer buff.*

PASCAL: *The language that's set the computer world ablaze. When you've programmed in Pascal you'll wonder why you ever used anything else. Powerful, elegant, yet easy to learn, it's a fascinating language designed to improve both programming and programs.*

With these exciting languages available instantly on ROM, the Language Lab gives your Electron two new, exciting and very different personalities. It's the best package for the Electron yet. No programmer will want to be without it. And it comes to you at LESS THAN HALF the usual price!

Normal price £197.90 ***Electron User price £89.95***

HELICOPTER RESCUE

By GEOFF SEARLE

HELICOPTER Rescue is a game based on the rescue of two men whose boat has become stranded on rocks. You, as the pilot of the helicopter, have to rescue the men and land them safely on the cliff-top.

It is not a fast-action game, but needs concentration and skill, especially when the wind speed increases, as it does each time you save the men.

At the beginning, brief instructions are given, and the Electron draws a random pattern of rocks and a cliff. It places a boat and the two men somewhere on the rocks. One of the men will always be in a position that can be reached by the helicopter.

The Electron then selects a wind-speed and direction – at first you will be told the wind direction, but later you will have to work it out for yourself.

You control the helicopter with the following keys:

Z	Turn left.
X	Turn right.
:	Increase power.
/	Decrease power.
Ctrl	Ascend.
Shift	Descend.
Q	Raise winch.
A	Lower winch.

First, you must position the helicopter over the boat – it must, obviously, be facing into the wind to hover, and the stronger the wind the more power will be needed to maintain the hover.

The wind gusts as well, so the power will have to be constantly juggled to keep the helicopter steady.

Go down as low as you can with the helicopter, without touching the rocks, and lower the winch.

When you touch one of the men with the end of the winch, he is secured and you can then winch him up into the helicopter, fly back to the cliff, and lower him on to the cliff-top before returning for the second man.

If the first one that you rescued was the only man you could reach, the second one will "climb up" to where the first one was.

If, during the rescue, the winch touches a rock, the winch breaks and is lost.

If the helicopter touches the rocks, it crashes, and if you try to move too fast with the winch down it becomes tangled in the rotors, and again the helicopter crashes – and you only get one chance!

Sometimes the men will be in a position which necessitates a long winch being used – for example, right at the foot of the cliffs.

Once the winch gets over a certain length, it starts to

PROCEDURES

heli_move	Calculates and prints new position of helicopter and checks to see if it has crashed.
winch	Calculates new position of winch and checks to see if it has rescued a man or touched a rock.
crash	Self-explanatory.
winch_break	Self-explanatory.
survivor	Places rescued man on bottom of winch.
rescued	Lands rescued man on cliff top.
title	Title page, controls and warnings.
rocks	Draws rocks and positions casualty.
chars	Defines characters and envelopes.

FLAGS

h%	Direction in which helicopter is flying (–1 or 1).
swing%	Direction in which winch is swinging (–1 or 1).
Wdir%	Direction of wind (–1 or 1).
sw%	Set if winch is to swing.
man%	Man on end of winch.
cr%	Helicopter crashed.
end%	End of game.
new%	New game.

VARIABLES

L%	Game level (in steps of 10).
A%, **B%**	Actual coordinates of helicopter.
A	Calculated x-coordinate of helicopter.
a%, **b%**	Previous coordinates of helicopter.
POW%	Power level.
HL$	Character string for left-hand side of helicopter.
HR%	Character string for right-hand side of helicopter.
H$	HL$ or HR$
WS%	Approximate wind-speed.
WI	Variation of wind from WS% due to gusting.
Wind%	Actual wind-speed.
WTX%, **WTY%**	Coordinates of top of winch.
WBX%, **WBY%**	Coordinates of bottom of winch in relation to helicopter.
WB%	Actual 'y'-coordinate of bottom of winch.
WBs%	Displacement of bottom of winch due to swinging.
wsmax	Maximum amount that winch can swing.
wistep	Amount that winch moves vertically.
wipt%	'y'-coordinate of top of winch as it falls.
RES%	Number of men rescued.
time%	Time to rescue both men.
low%	Lowest time so far.
mrx%, **mry%**	Coordinates of rescued man on end of winch.
cpx%, **cpy%**	Coordinates of casulty's initial position.
CX%	'x'-coordinate of boat.
RX%, **RY%**, **INC%**	Used in drawing rocks.

swing, and the stronger the wind, the more it swings.

This can make the rescue very difficult, especially if the casualties are down between two rocks.

On the other hand, sometimes the only way that the men can be reached is by swinging the winch on purpose.

Another way in which men in apparently inaccessible positions can be reached is by turning the helicopter around, as this slightly alters the position of the winch.

The helicopter can fly right up to the edges of the screen, but if you stay at either edge for too long, the helicopter will crash.

If you crash or the winch breaks, you are given the opportunity to play again, with the wind-speed roughly the same as before.

If you successfully rescue both men, you can play again but this time the wind-speed increases.

When the wind gets up above about 60 knots, you really have to have your wits about you. Actually, you can cheat – just alter the value of L% in line 90 to roughly the wind-speed you want.

At the end of each game you successfully complete, you will be told how long it took you to rescue the two men, and the lowest time so far. A time under 100 seconds is good – under a minute is excellent.

Rescue will take quite a while to get used to, but don't despair – RAF pilots spend months training . . .

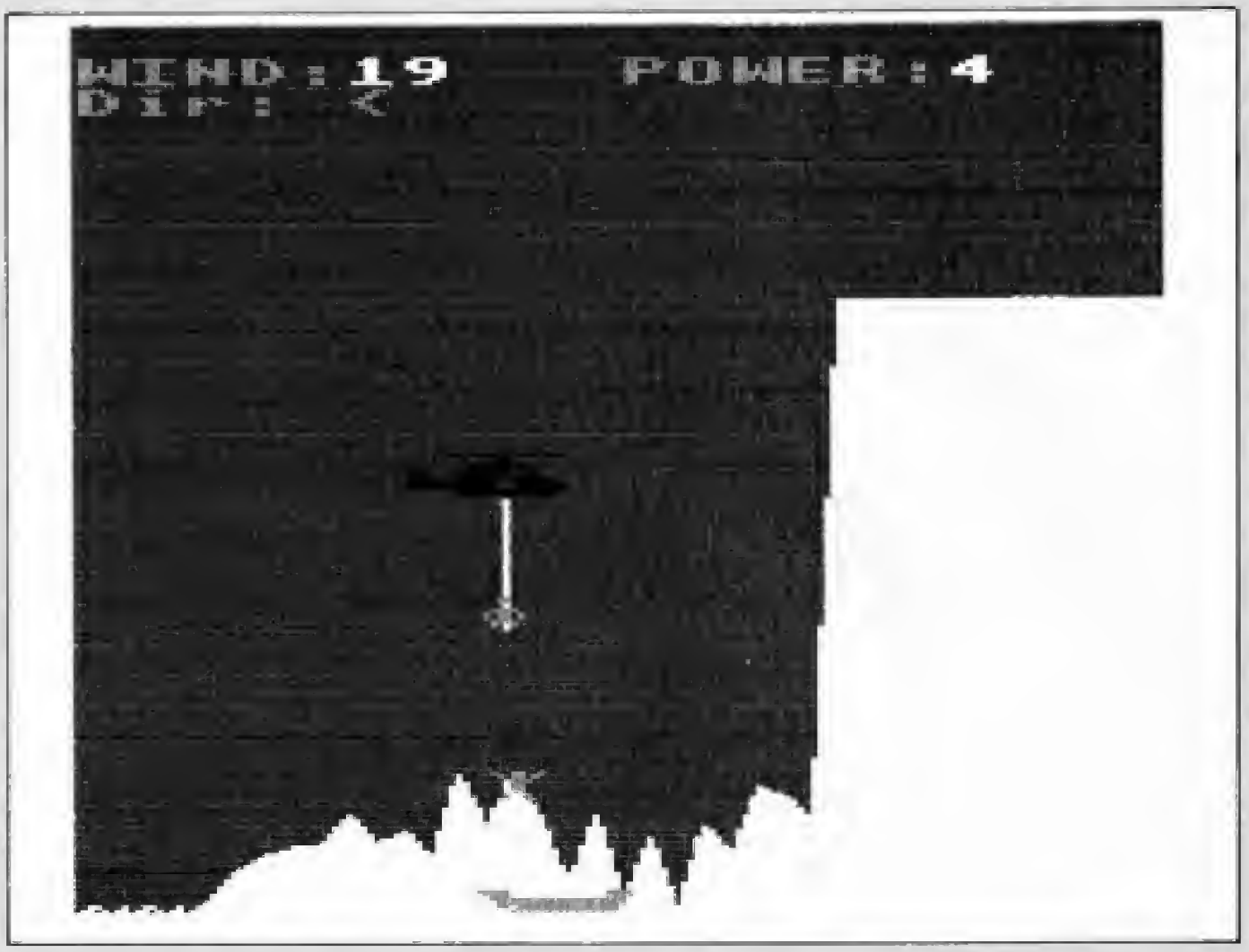
Once you do get the hang of controlling the helicopter, try to increase your speed.

Full listing starts on Page 40

From Page 39

```
  10 REM R E S C U E
  20 REM By Geoff Searle
  30 REM (c) Electron User
  40 ON ERROR GOTO1960
  50 MODE1
  60 PROCtitle
  70 MODE2
  80 PROCchars
  90 LX=10:lowX=300
 100 COLOUR134:CLS
 110 POWX=14:A=10:AX=10:BX=5:WSX=LX+RND(20):WI=0:H$=HL$:hX=-1:WTXX=704:WTYX=811:WBXX=704:WBYX=0:WBX=811:RESX=0:swingX=0:WBsX=0:WindX=0:aX=0:bX=0
 120 crX=FALSE:manX=FALSE:resX=FALSE:endX=FALSE:newX=FALSE
 130 IF WSX>140 THEN WSX=140
 140 PROCrocks
 150 wX=RND(2):IF wX=1 THEN WdirX=1 ELSE WdirX=-1
 160 COLOUR1:PRINTTAB(0,2);"WIND:";TAB(10,2);"POWER:"
 170 IF LX>40 GOTO190
 180 IF wX=1 THEN PRINTTAB(0,3);"Dir: >" ELSE PRINTTAB(0,3);"Dir: <"
 190 TIME=0
 200 SOUND0,1,4,255
 210 REM  LOOP START
 220 PROCheli_move
 230 IF endX GOTO270
 240 IF newXTHENGOTO 100
 250 GOTO220
 260 REM  LOOP END
 270 VDU4:PRINTTAB(5,8);"WELL DONE!"
 280 timeX=INT(TIME/100)
 290 PRINT'" YOU HAVE RESCUED"'" BOTH MEN IN ";timeX;'SPC(4);" SECONDS"
 300 IF timeX<lowX PRINT'" THIS IS THE LOWEST"'" TIME SO FAR"':lowX=timeX:GOTO320
 310 PRINT'"THE LOWEST TIME"'" SO FAR IS ";lowX;'" SECONDS"
 320 PRINT'"   PRESS SPACE "'" FOR A NEW GAME"
 330 REPEATUNTILGET=32
 340 LX=LX+10:aX=0:bX=0:GOTO100
 350 REM  PROCEDURES
 360 DEF PROCheli_move
 370 IF INKEY(-98)THENH$=HL$:hX=-1
 380 IF INKEY(-67)THENH$=HR$:hX=1
 390 COLOUR6:PRINTTAB(aX,bX);H$
 400 IFPOINT(AX+64,1023-(BX+32)-52)<>6ORPOINT(AX+64+8,1023-(BX+32)-56)<>6ORPOINT(AX+64+40,1023-(BX+32)-56)<>6ORPOINT(AX+64+72,1023-(BX+32)-56)<>6ORPOINT(AX+64+104,1023-(BX+32)-56)<>6 THEN crX=TRUE
 410 IFPOINT(AX+64+100,1023-(BX+32)-52)<>6 ORPOINT(AX+64+136,1023-(BX+32)-56)<>6 THENcrX=TRUE
 420 COLOUR3:PRINTTAB(AX,BX);H$:aX=AX:bX=BX
 430 IF crX THENPROCcrash:crX=FALSE
 440 IF newX GOTO 580
 450 PROCwinch
 460 IF INKEY(-73)POWX=POWX+2:IF POWX>50THEN POWX=50
 470 IF INKEY(-105)POWX=POWX-2:IF POWX<2THENPOWX=2
 480 IF INKEY(-2)THENBX=BX-1:IF BX<4THENBX=4
 490 IF INKEY(-1)THENBX=BX+1
 500 WI=WSX/150+RND(50))+(LX/2)/(100+RND(50))
 510 A=A+(WI*WdirX)+((POWX/15)*hX)
 520 AX=INT(A)
 530 IF AX>17 THENAX=17:IF A>22 THEN PROCcrash
 540 IF AX<0THENAX=0:IF A<-4 THENPROCcrash
 550 WindX=WSX+INT(10*WI)
 560 COLOUR0:PRINTTAB(5,2);WindX;" ";
 570 PRINTTAB(16,2);POWX;" ";
 580 ENDPROC
 590DEF PROCwinch
 600wsmax=0:swX=FALSE
 610MOVEWTXX,WTYX:GCOL0,6:DRAWWBXX,WBX
 620IF H$=HL$ THEN WTXX=AX+64+64 ELSE WTXX=AX+64+120
 630WBXX=WTXX
 640WTYX=1023-(BX*32)-52
 650IF WindX>50 THEN wsmax=wsmax+8:swX=TRUE
 660IF WindX>100 THEN wsmax=wsmax+8:swX=TRUE
 670IF WBYX>150 THEN wsmax=wsmax+8:swX=TRUE
 680IF NOT swX THENGOTO 710
 690WBsX=WBsX+swingX:IF ABS(WBsX)>=wsmax THEN swingX=-swingX
 700WBXX=WBXX+WBsX
 710IF WTXX>RXX AND WBX>690 THEN wistepX=8 ELSE wistepX=32
 720IF WBX<250 THEN wistepX=8
 730IF INKEY(-17)WBYX=WBYX-wistepX:IF WBYX<0THENWBYX=0
 740IF INKEY(-66)WBYX=WBYX+wistepX
 750WBX=WTYX-WBYX:IF WBX<50THENWBX=50
 760GCOL0,0:MOVEWTXX,WTYX:DRAWWBXX,WBX
 770IF manX THENVDU5 ELSE GOTO800
 780GCOL0,6:MOVEmrxX,mryX:PRINTCHR$(241)
 790mrxX=WBXX-32:mryX=WBX+32:MOVEmrxX,mryX:GCOL0,1:PRINTCHR$(241):VDU4:GOTO810
 800IF POINT(WBXX,WBX-4)=5 ORPOINT(WBXX,WBX-8)=5 THEN PROCsurvivor:GOTO830
 810IF POINT(WBXX,WBX-8)=0 AND manX AND WBX>690 THENPROCrescued:GOTO830
 820IF POINT(WBXX,WBX-8)=0 THENPROCwinch_break
 830VDU4:ENDPROC
 840DEF PROCcrash
 850 *FX21,4
 860SOUND0,3,1,255
 870*FX9,2
 880*FX10,2
 890COLOUR9:PRINTTAB(AX,8X);H$
 900COLOUR1:PRINTTAB(3,10);"CRASHED!!"'" PRESS SPACE FOR"'"  ANOTHER GAME"
 910*FX15,1
 920REPEATUNTIL GET=32
 930*FX21,4
 940aX=0:bX=0
 950newX=TRUE:ENDPROC
 960DEF PROCwinch_break
 970 *FX21,4
 980VDU5
 990IF WBYX>500 D=3
1000IF WBYX<501 D=2
1010IF WBYX<250 D=1
1020 ENVELOPE2,D,-1,0,0,25,0,0,126,0,0,-126,126,126
1030SOUND1,2,0,160
1040wiptX=WTYX
1050MOVE WTXX,WTYX:GCOL0,6:DRAWWBXX,WBX
1060REPEAT
1070MOVE WTXX,wiptX:GCOL0,0:PLOT21,WBXX,WBX-32:IF manXTHEN MOVE WBXX-32,WBX+32:GCOL0,1:VDU234
1080GCOL0,6:MOVE WTXX,wiptX:DRAWWBXX,WBX
1090wiptX=wiptX-8
1100UNTILwiptX<WBX+28
1110*FX15,1
1120*FX21,5
1130VDU4:COLOUR1:PRINTTAB(1,6);"WINCH BROKEN!"'" PRESS SPACE FOR"'" ANOTHER GAME"
1140REPEATUNTILGET=32
1150aX=0:bX=0:newX=TRUE:ENDPROC
1160DEF PROCsurvivor
1170VDU5
1180GCOL0,0:MOVEcpxX+72,96:PRINTCHR$(242)
1190MOVEWTXX,WTYX:GCOL0,6:DRAWWBXX,WBX
1200WBYX=WTYX-cpyX-8:GCOL0,0:MOVEWTXX,WTYX:DRAWWBXX,WTYX-WBYX:GCOL0,1:MOVEWBXX-32,WTYX-WBYX+32:PRINTCHR$(241)
1210mrxX=WBXX-32:mryX=WTYX-WBYX+32
1220IF RESX=1THENGCOL0,6:MOVE cpxX,cpyX:PRINTCHR$(234):GOTO1240
1230GCOL0,5:MOVEcpxX,cpyX:PRINTCHR$(234);
1240manX=TRUE
1250*FX12,0
1260*FX15,1
1270VDU4
1280FORtX=1TO100:NEXT
1290ENDPROC
1300DEF PROCrescued
1310SOUND 1,4,80,7:SOUND 1,4,60,3:SOUND 1,4,0,1:SOUND 1,4,60,3:SOUND 1,4,68,7:SOUND 1,4,60,14:SOUND 1,4,76,7:SOUND 1,4,80,9
1320manX=FALSE:WBYX=0
1330VDU5:GCOL0,6:MOVE WTXX,WTYX:DRAWWBXX,WBX:MOVEmrxX,mryX:GCOL0,1:VDU241
```

```
 1340RESX=RESX+1:IF RESX=2T
HENendX=TRUE
 1350WBX=WTYX:WBYX=0
 1360 FORT=1TO1100:NEXT:IF
NOT endX SOUND0,1,4,255
 1370ENDPROC
 13800EF PROCtitle
 1390VDU19,0,4;0;0;0;
 1400COLOUR130:CLS
 1410VDU23,1,0;0;0;0;
 1420COLOUR1:PRINTTAB(10,2)
;"R E S C U E"'TAB(10);"===
========"'TAB(28)"By G.Sear
le"
 1430COLOUR4:PRINT''"CONTRO
LS:"" Turn left
        Z"" Turn right
             X"" Ascend
                CTRL"" De
scend                  SHIFT
"" Winch down
     A"" Winch up
          Q"
 1440PRINT" Increase power
           :"" Decrease
power              /"
 1450COLOUR1:PRINT"WARNING
S:"
 1460COLOUR4:PRINT"If helic
opter touches rocks, it cra
shes.";"If winch touches ro
cks, it breaks."'"If helico
pter stays at edge too long
, itcrashes."'"Winch swings
 if long or in high winds."
'"Fly into wind to hover."
 1470PRINT"WINCH rescued ma
n onto cliff."'"If helicopt
er moves too much with winc
h down, the winch tangles i
n rotors."
 1480PRINT'''"      Press SP
ACE to start":REPEATUNTILGE
T=32:VDU19,0,0;0;0;0;::ENDPR
OC
 1490 DEF PROCrocks
 1500 cliff=FALSE
 1510 GCOL0,0:RX%=0:RY%=RND
(50)
 1520 PLOT4,0,0:PLOT4,0,RY%
 1530 REPEAT
 1540 PLOT85,RX%,RY%:PLOT85
,RX%,0
 1550 S=RND(2):IF S=1 THEN
INC%=1 ELSEINC%=-1
 1560 INC%=INC%*RND(100)
 1570 RX%=RX%+32:RY%=RY%+IN
C%
 1580 IF RY%<0 THENRY%=RND(
100)
 1590 IF RY%>200 AND RX%<85
0 THENRY%=RND(200)
 1600 IF RY%>200 AND RX%>85
0 THENRY%=700:cliff=TRUE
 1610 UNTILRX%>1000 OR clif
f
 1620 PLOT85,RX%,700:PLOT85
,1279,0:PLOT85,1279,700
 1630 GCOL0,4:PLOT4,0,0:PLO
T4,0,50:PLOT85,1279,0:PLOT8
5,1279,50
 1640 VDU5
 1650 MOVE0,50:PRINT"******
**************";:REM 20 OF
*
 1660 pos=FALSE:r%=200
 1670 CX%=RND(RX%)
 1680 IF CX%<64 THENCX%=64:
GOTO1700
 1690 IF CX%>RX%-128 THENCX
%=RX%-128
 1700 GCOL0,1:MOVECX%,90:PR
INTCAS$
 1710 GCOL0,5
 1720 REPEAT:IF POINT(CX%+4
8,r%-32)=0 ORPOINT(CX%+48,r
%-32)=1 THEN MOVE CX%+16,r%
:PRINTCHR$(234):cpx%=(CX%+1
6):cpy%=r%:pos=TRUE
 1730 r%=r%-8
 1740 UNTILpos
 1750 MOVE CX%+80,96:PRINTC
HR$(234);
 1760 VDU4
 1770 ENDPROC
 1780 DEF PROCchars
 1790 VDU23,1,0;0;0;0:
 1800 ENVELOPE 1,1,1,-1,0,3
,3,0,126,0,0,-126,126,126:E
NVELOPE3,2,1,-1,0,6,6,0,126
,0,0,-126,126,126
 1810 REM Helicopter left s
ide
 1820 VDU23,225,0,255,0,0,7
,15,12,63,23,226,0,255,192,
192,252,252,255,255,23,227,
0,192,0,0,0,0,14,14,23,228,
126,254,254,127,63,0,0,0,23
,229,63,63,63,255,254,0,0,0
,23,230,254,254,254,0,0,0,0
,0
 1830 REM Helicopter right
side
 1840 VDU23,235,0,3,0,0,0,0
,112,112,23,236,0,255,3,3,6
3,63,255,255,23,237,0,255,0
,0,224,48,48,240,23,238,127
,127,127,0,0,0,0,0,23,239,2
52,252,252,255,127,0,0,0,23
,240,126,127,127,254,252,0,
0,0
 1850 REM Boat
 1860 VDU23,231,0,0,248,120
,63,31,27,7,23,232,0,0,0,0,
255,255,255,255,23,233,0,0,
30,30,252,252,248,248
 1870 REM Men
 1880 VDU23,234,153,90,60,2
4,24,24,36,102,23,242,157,2
54,254,254,254,255,255,255,
23,241,24,24,0,60,90,90,60,
36
 1890 REM Explosion
 1900 VDU23,250,130,68,33,0
,9,32,68,129,23,251,74,32,3
7,0,82,4,82,81,23,252,137,3
4,2,84,36,138,32,153
 1910 HL$=CHR$(225)+CHR$(22
6)+CHR$(227)+CHR$(8)+CHR$(8
)+CHR$(8)+CHR$(10)+CHR$(228
)+CHR$(229)+CHR$(230)
 1920 HR$=CHR$(235)+CHR$(23
6)+CHR$(237)+CHR$(8)+CHR$(8
)+CHR$(8)+CHR$(10)+CHR$(238
)+CHR$(239)+CHR$(240)
 1930 CAS$=CHR$(231)+CHR$(2
32)+CHR$(233)
 1940 ENVELOPE5,2,1,1,-1,1,
1,2,126,0,0,-126,126,126
 1950 ENDPROC
 1960 MODE6:PRINT:REPORT:PR
INT" at line ";ERL
```



This listing is included in this month's cassette tape offer. See order form on Page 61.

21st. Software (BBC VERSIONS AVAILABLE)

PRESENTS A SELECTION OF HARDWARE, UTILITIES & SOFTWARE FOR THE ELECTRON

PLAY GAMES AT
YOUR OWN SPEED
OR FREEZE FRAME
WITH
"SLOMO"
R.R.P. £14.95 OUR PRICE £13.45
Fully compatible with the Plus 1

3 ACORNS PACK
3 cassettes: Boxer, Starship Command, Desk Diary
R.R.P. £20.97 OUR PRICE £10.45

PLUS3 SOFTWARE ACORNSOFT GAMES DISC
(Contains: Maze, Firebug & Planetoids)
R.R.P. £19.75 OUR PRICE £17.95
Database R.R.P. £19.75 OUR PRICE £17.95

TALENT 3 PACK
2 Cassettes: WEST (Text adv.) £7.95
LASER REFLEX (Esc. Arc. game) £4.95
R.R.P. £12.90
OUR PRICE £4.95

FIRST BYTE JOYSTICK INTERFACE
Includes conversion tape £19.95
+ Quickshot II Joystick £12.99
TOTAL £32.94 OUR PRICE £7.99

SUPERIOR SOFTWARE'S
SMASH PACK
3 CASSETTES
Canabag (Stamp the Centipede)
Percy Penguin (Crush the Snowbees)
Zany Kong Jnr. (Climb the vine leaves – watch out for the snappers!
Fantastic value
All 3 for just £7.95!

NEW
SUPERIOR'S
CITADEL
R.R.P. £9.95 OUR PRICE £8.95

NEW
AARDVARK'S
FRAN
R.R.P. £7.90 OUR PRICE £6.90

POWERSOFT ROM JOYSTICK
INTERFACE
R.R.P. £24.95 OUR PRICE £22.65

FIRST BYTE JOYSTICK I/FACE
Includes conversion tape
R.R.P. £19.95 OUR PRICE £18.25

SLOGGER BUSINESS PACK

Rombox MK II	£44.95
Starword (Word Processor)	£34.50
Starstore (Database)	£29.95
	£109.40

OUR PRICE = £89.00 only

SLOGGER ROMBOXES
ROMBOX R.R.P. £44.95 OUR PRICE £41.95
ROMBOX P
Built in Centronics Printer Interface + Free
Printer ROM R.R.P. £69.95 OUR PRICE £64.95

VINE MICROS ADDCOM
R.R.P. £28.60
OUR PRICE £26.00

SLOGGER SOFTWARE ROMS
ELKMAN R.R.P. £17.50 OUR PRICE £15.75
STARMON (machine code monitor)
R.R.P. £22.50 OUR PRICE £20.25
T2P3 Tape to Disc R.R.P. £19.95
OUR PRICE £18.65
STARWORD 16K ROM
R.R.P. £34.50 OUR PRICE £31.75
STARGRAPH GRAPHICS ROM
R.R.P. £21.95 OUR PRICE £19.95

KANSAS CITY
"GOLD" PACK
3 Cassettes. Multi Level.
Excellent Full Colour Graphics
1 SNAKE (ARC Repeater)
2 MANIC MOWER (Protect Your Toes!)
3 CAVEMAN (Dig for Gold & Diamonds)
R.R.P. £20.85 OUR PRICE £6.95!!!

	R.R.P	OUR PRICE
A'N'F		
Bargain Pack	14.95	12.95
AARDVAAK		
Zalaga	7.90	6.90
A.C. PRODUCTS		
Advanced Disc Toolkit Rom	34.50	32.20
ACORNSOFT		
Elite	12.95	11.85
Magic Mushrooms	11.95	10.85
Snooker	6.99	5.99
Sphinx Adventure	6.99	5.99
Starship Command	6.99	5.99
Arcadians	6.99	5.99
Free Fall	6.99	5.99
Meteors	6.99	5.99
Maze	6.99	5.99
Boxer	6.99	5.99
Philosophers Quest	6.99	5.99
Snapper	6.99	5.99
Monsters	6.99	5.99
Hopper	6.99	5.99
Chess	6.99	5.99
Business Games	6.99	5.99
Desk Diary	6.99	5.99
Firebug	6.99	5.99
Tomb	16.85	15.35
Trip	16.85	15.35
Personal Money Management	6.99	5.99
Tree of Knowledge	6.99	5.99
View Rom Cartridge	29.99	27.65
Viewsheet Rom Cartridge	29.99	27.65
Hopper ROM (Cart)	14.95	13.45
Snapper ROM (Cart)	14.95	13.45
BSO Pascal ROM Cart	59.80	54.80
Lisp ROM Cart.	29.95	27.65
ACORNSOFT/BES		
Happy Letters	8.95	7.95
Timeman Doo	8.95	7.95
Happy Numbers	8.95	7.95
Wordhang	8.95	7.95
Digrey	9.95	8.95
ACORN/LINKWORD		
German	14.95	12.95
French	14.95	12.95
Italian	14.95	12.95
Spanish	14.95	12.95
ACORN/BERG		
Royal Quiz	12.95	11.75
Science Fiction Quiz	12.95	11.75
ADDICTIVE GAMES		
Football Manager	8.95	7.95
Bullin	9.95	8.95
ADVENTURE INT		
Gremlins	7.95	6.95
The Hulk	7.95	6.95
Spiderman	7.95	6.95
Adventureland	7.95	6.95
The Count	7.95	6.95
Mystery Funhouse	7.95	6.95
Robin of Sherwood	7.95	6.95
Pyramid of Doom	7.95	6.95
Sorcerer Claymorgue	7.95	6.95
Strange Odyssey	7.95	6.95
Pirate Adventure	7.95	6.95

	R.R.P.	OUR PRICE
ALLIGATA		
Blagga	7.95	6.95
Bumper Bundle	7.95	6.95
Contract Bridge	9.95	8.95
Guardian	7.95	6.95
Tarzan Boy	7.95	6.95
Nightworld	7.95	6.95
ATARISOFT		
Pinbotics	9.95	8.95
ANIROG		
Jump Jet	9.95	8.95
A.S.K.		
Number Painter	9.95	8.95
Number Gulper	9.95	8.95
Best 4 English	19.95	18.45
Best 4 Maths	19.95	18.45
COMSOFT		
Serpents Lair	4.95	3.95
Playbox	4.95	3.95
S.A.S. Commando	4.95	3.95
C.O.S.		
Brides Barrage	7.95	6.95
C.R.L.		
Fast Match	7.95	6.95
DR. SOFT		
747 Flight Sim	7.95	6.95
Golf	7.95	6.95
Phantom Combat	9.95	8.95
D.A.C.C.		
Flight Sim 747 (DACC)	9.95	8.95
Sonic Generator	9.95	8.95
Bobby Charlton Soccer	11.95	10.95
DURELL MARTECH		
Geoff Capes	8.95	7.95
Minechalk	6.95	5.95
Brian Jacks Superstars	7.95	6.95
Combat Lynx	8.95	7.95
Eddie Kidd Jump	7.95	6.95
Gribbine's Castle	7.95	6.95
EDGE COMPUTERS		
Micrograf	9.95	8.95
ENGLISH SOFTWARE		
Jet Boot Jack	7.95	6.95
Kissin' Cousins	4.95	4.25
EPIC SOFTWARE		
The Wheel of Fortune	9.95	8.45
Castle Frankenstein	7.95	6.55
The Quest of the Holy Grail	7.95	6.55
Kingdom of Klein	7.95	6.55
GLOBAL		
Dp. Caretaker (Cleaning kit)	10.95	9.95
GOLEM LTD		
Education I	8.00	7.00
Education II	8.00	7.00
Fun with Words	8.00	7.00
Fun with Numbers	8.00	7.00
Jigsaw	8.00	7.00
HEWSON		
Heathrow A.T.C.	7.95	6.95
HIGHLIGHT SOFT		
Reading Pack I	7.95	6.95
Reading Pack II	7.95	6.95
Entertaining English	7.95	6.95
Reading Pack 3	7.95	6.95

	R.R.P	OUR PRICE
Reading Pack 4	7.95	6.95
HOLEY SOFTWARE		
Gamemaker Mode 2	12.00	10.65
Gamemaker Mode 5	12.00	10.65
ICON SOFTWARE		
Caveman Capers	7.95	6.95
Dream Mania	7.95	6.95
Bug Eyes	7.95	6.95
INCENTIVE		
Confusion	6.95	5.95
INTERCEPTOR		
Tales Arabian Knights	6.00	5.00
KANSAS CITY		
Ring of Tuba	9.50	6.50
Moon Buggy	6.95	5.95
KOSMOS SOFTWARE		
French Mistress A or B	8.95	7.95
German Master A or B	8.95	7.95
Spanish Tutor A or B	8.95	7.95
Answer Back Jun. or Sen.	9.95	8.95
Answer Back Sport	9.95	8.95
Identify Europe	7.95	6.95
L.C.L.		
Micro Maths (24 progs 'O' Level)	24.50	21.50
Micro English (24 progs 'O' Level)	24.50	21.50
Mega Maths (A Level)	24.50	21.50
LONGMAN'S SOFTWARE		
First Moves (Chess 8 – 1	9.95	8.95
Master Key (Typing Tutor)	9.95	8.95
MACSEN SOFT		
Black Bastion	7.95	6.95
MELBOURNE HOUSE		
Classic Adventure	6.95	5.95
Hampstead House	6.95	5.95
Terrormolinos	6.95	5.95
M.C. LOTHLORIEN		
Redcats	6.95	5.95
Paras	6.95	5.95
Johnny Reb	6.95	5.95
Special Operations	6.95	5.95
Waterloo	9.95	8.95
Battlezone	6.95	5.95
MICROBYTE		
Mekata Raiders	6.95	5.95
Ex*Bert	4.95	3.95
Draughts + Reversi	4.95	3.95
Pinball	4.95	3.95
MICRODEAL		
Space Shuttle	8.00	7.00
MP SOFTWARE		
Sadam Castle	7.50	6.50
Valley of the Kings	7.50	6.50
Frozen Wood	7.50	6.50
Woodland Terror	7.50	6.50
Blue Dragon	7.50	6.50
Survivor	7.50	6.50
MICROPOWER		
Ghouls	7.95	6.95
Jet Power Jack	6.95	5.95
Pogoman	6.95	5.95
Swag	6.95	5.95
Gauntlet	6.95	5.95
Chess	7.99	6.95
Killer Gorilla	7.95	6.95

	R.R.P	OUR PRICE
Moonraider	7.95	6.95
U.K.B.	7.95	6.95
Felix Evil Weevils	6.95	5.95
Felix in Factory	7.95	6.95
Felix Fruit Monsters	7.95	6.95
Galactic Commander	7.95	6.95
Cybertron	7.95	6.95
Shock Car	6.95	5.95
Whack Ball	6.95	5.95
Rubble Trouble	6.95	5.95
Adventure	7.95	6.95
MIRRORSOFT		
Hamet Strike Force	9.95	8.95
Count with Oliver	7.95	6.95
Look Sharp	7.95	6.95
First Steps	8.95	7.95
Here/There with Mr. Men	7.95	6.95
Quick Thinking	8.95	5.95
WORD GAMES		
Mastermind	9.95	8.95
Starspeller	9.95	8.95
OASIS		
Aces High	14.95	12.95
OCEAN		
Hunchback	6.90	5.90
SALAMANDER		
737 Flight Sim	9.95	8.95
Graphics System	9.95	8.95
SHARDS		
Empire	6.95	5.95
Pellegrines Diary	7.95	6.95
Mystery of Java Star	7.95	6.95
Galilee	6.95	5.95
Woodbury End	9.95	8.95
Whoopee	6.95	5.95
SOFTWARE INVASION		
3D Bomb Alley	7.95	6.95
Gunsmoke	7.95	6.95
Blockaway	7.95	6.95
Super Pool	7.95	6.95
Votex	7.95	6.95
Chopbuster	7.95	6.95
Stairway to Hell	13.95	12.95
SLOGGER		
Dogfight (2 player joystick) Key spt	7.95	3.00
SQUIRRELSOFT		
Supergolf	7.50	6.50
Trafalgar	8.00	7.00
Polar Perils	7.95	6.95
SUPERIOR SOFTWARE		
Mr Wiz	7.95	6.95
Smash and Grab	7.95	6.95
Overdrive	7.95	6.95
Tempest	9.95	8.95
Repton	9.95	8.95
Death Star	9.95	8.95
Repton 2	9.95	8.95
TYNESOFT		
Supercar (Adv)	£7.95	£6.95
U.S. GOLD		
Beach Head	£8.95	£7.95
VISIONS/CSM		
Snooker	8.95	7.95
Daredevil Dennis	7.95	6.95
Sim	7.95	6.95

BOX OF 10 C15 COMPUTER CASSETTES £4.50 inc VAT & P&P.

ALL PRICES ARE INCLUSIVE OF VAT AND CARRIAGE
OVERSEAS ORDERS PLEASE ADD £1.00 THEN 50p FOR 2ND and 3RD ITEMS EACH

We guarantee all titles are originals. We offer a quick and reliable service. Most Electron and BBC titles are available on request with the guarantee of at least £1.00 off the R.R.P.

Please send me:

	Cost
1. _____	£
2. _____	£
3. _____	£
4. _____	£
TOTAL	£

Please make cheques payable to:
21st Software.

Send orders to:
21st SOFTWARE
20 Woodside Drive
High Lane, Stockport SK6 8HU
Tel: Disley (06632) 3160.

Name .....
Address .....
.....
.....
Post Code .....
Tel. No. .....

Part Five

*FX OS CALLS

Procedures to pep up screen displays

By JOHN WOOLLARD

THIS month we're going to take a look at the FX calls concerned with graphics and screen displays – and I'll be presenting you with some procedures that you may find useful in your own programs. Let's make a start with the colours.

In the first article of this series I showed how *FX9 and *FX10 are used to change the on/off rate of the flashing colours. We can now extend the technique to read the colour flash rates. This is done using osbyte calls 194 and 195, or *FX194 and *FX195 if you like. However to read information back we have to enter the procedure shown in Program I and use that to make the calls.

Changing the flash rates is easy: simply decide how long you want the colour to stay on and how long it should be off. (The time is measured in fiftieths of a second.)

Run program I and put some flashing writing on the screen by entering:

```
MODE 2
COLOUR 8
LIST
```

The colour space period can be changed to 5 seconds using PROCfx(194,250,0).

The colour mark state can be set to 1/50 second using PROCfx(195,1,0).

To set the colour mark state to infinity you should enter PROCfx(195,0,0). This stops the flashing completely.

We can use the same procedure to read the present flash rates. We make the call with the *X* value equal to zero and the *Y* value equal to 255.

That combination of numbers leaves the flash rates unchanged but their values are returned in the variable y%.

Try changing the flash rates, then investigate your changes using:

```
PROCfx(194,0,255):PRINTy%
```

and

```
PROCfx(195,0,255):PRINTy%
```

Two other associated calls are *FX193 and *FX154. *FX193 is used to determine how long it is until the next change in colour.

```
PROCfx(193,0,255)
PRINTx%" 1/50 secs"
```

*FX154 is used to reset the flash sequence to the start. Visually it's not important if the default setting of half a second on then half a second off is being used.

However if the periods are set longer then it's an improvement if the flashing cycle starts the moment the text or graphics appears.

I tend to use flashing colours when important messages are displayed.

It looks less professional if the message appears for the merest fraction of a second before vanishing. By resetting the flash cycle with *FX154 the flashing message begins with a more even appearance.

Moving on, have you ever noticed that some arcade games start with the screen completely black for a few seconds and then the graphic display appears instantaneously?

The effect is achieved by turning all colours to black, drawing the graphics and then turning the colours on again.

You've probably guessed that an osbyte call does all that.

*FX115,1, turns all colours black, *FX115,0 turns them on again.

Program II contains a useful procedure for producing circles quickly. *FX115 has been used so that during the drawing process the screen remains completely black. The 16 circles appear together when they've all been drawn.

One technique that I've used to generate a striking display is to repeatedly print text or plot graphics, then immediately to clear the screen.

This produces a flickering effect, not the steady flash that you'd expect. This is because the text does not appear the

——►

```
10REM Program I
20REM
30REM
40DEFPROCfx(aa%,xx%,yy%)
50o%=&FFF4
60A%=aa%
70X%=xx%
80Y%=yy%
90r%=USR(o%)
100a%=r%AND&FF
110x%=(r%AND&FF00)DIV&100
120y%=(r%AND&FF0000)DIV&1
0000
130r%=(r%AND&FF000000)DIV
&1000000
140ENDPROC
```

Program I

```
10REM Program II
20REM
30REM
40MODE2
50*FX115,1
60FORk%=1TO16
70GCOL1,k%
80MOVE640,1000
90PROCcircle(k%*48)
100NEXT
110*FX115,0
120END
130DEFPROCcircle(size%)
140LOCALk%,xdata%,ydata%
150size%=size%DIV50
160RESTORE220
170FORk%=0TO41
180READxdata%,ydata%
190PLOT1,xdata%*size%,yda
ta%*size%
200NEXT
210ENDPROC
220DATA0,0,7,-1,6,-2,7,-3
,7,-4,6,-5,5,-6,4,-7,3,-7,2
,-8,-1,-7,-2,-6,-3,-7,-4,-7
,-5,-6,-6,-5,-7,-4,-7,-3,-8
,-2,-7,-1,0
230DATA0,-7,1,-6,2,-7,3,-
7,4,-6,5,-5,6,-4,7,-3,7,-2,
8,-1,7,2,8,3,7,4,7,5,6,6,5,
7,4,7,3,8,2,7,1,0,0,2,0
```

Program II

From Page 43

moment the computer carries out a print instruction.

The screen is updated 50 times a second. You might think that that is quite frequent, but it's possible for the computer to print on to the screen and then clear it between updates of the screen. Consequently, we don't see it.

Enter Program III into your micro but leave out line 60, the *FX19 call. When it is run you will see that the asterisk does not flash in a steady fashion.

The osbyte call makes the computer wait until the next

```
10REM Program III
20REM
30REM
40MODE2
50REPEAT
60*FX19
70CLS
80PRINTTAB(4,10)"*"
90UNTILFALSE
100END
```

Program III

update of the screen is due before it can procede.

As a result the asterisk is printed and then cleared in the same part of the screen update circle every time. The flicker becomes very even.

Another way of generating an even flicker is to use the osbyte call mentioned earlier, *FX115.

Instead of clearing the screen and reprinting or replotting repeatedly, the display is made up only once.

The palette – that's the range of colours available – is then switched to black and back again.

Program IV illustrates the technique using INKEY statements to control the flash rate.

It should be noted that *FX115 only operates in the high resolution modes.

Talking of modes, did you know that the computer can discover what mode it's in? You may ask, why do you need to find out?

I'm a great believer in building up a bank of procedures that can be used as and when necessary in any program.

Program V contains a

```
10REM Program IV
20REM
30REM
40MODE2
50PRINTTAB(4,10)"Electro
n User"
60REPEAT
70*FX115,1
80inkey%=INKEY(20)
90*FX115,0
100inkey%=INKEY(20)
110UNTILINKEY(-74)
120END
```

Program IV

procedure for putting a border around the screen. Before it can do that it needs to know the size of the screen, which is determined by the mode in use.

Program V also contains a function that returns that value.

The osbyte call used has *A%=135*, *X%=0* and *Y%=0*. It's the same call that's used to read a character at the text cursor position. The mode value is returned in *Y%*.

Lines 50 to 120 calculate the value of the width and height of the screen from the mode value.

There are two other calls associated with screen modes. *FX132 returns the value of HIMEM. That value varies with the mode that's in use.

It marks the highest point of space available for a user's program.

*FX133 will read the value of HIMEM for any given mode.

```
10REM Program V
20REM
30REM
40mode%=FNmode
50width%=40
60depth%=32
70IFmode%=7THENdepth%=25
:REM BBC only
80IFmode%=6THENdepth%=25
90IFmode%=5THENwidth%=20
100IFmode%=3THENdepth%=25
:width%=80
110IFmode%=2THENwidth%=20
120IFmode%=0THENwidth%=80
130PROCborder(width%,dept
h%)
140END
150DEFFNmode
160A%=135
170Y%=0
180X%=0
190r%=USR(&FFF4)
200=(r%AND&FF0000)DIV&100
00
210DEFPROCborder(x%,y%)
220LOCALk%
230PRINTTAB(0,0);
240FORk%=1TOy%
250PRINTSTRING$(x%,"*");
260NEXT
270FORk%=1TOy%-5
280PRINTTAB(2,1+k%)STRING
$(x%-4," ");
290NEXT
300ENDPROC
```

Program V

The following instructions use the procedure found in Program I:

```
PROCfx(132,0,0):PRINT"HIMEM
=&";~y%*256+x%
PROCfx(133,1,0):PRINT"MODE1
_HIMEM=&";~y%*256+x%
PROCfx(133,6,0):PRINT"MODE6
_HIMEM=&";~y%*256+x%
```

The next program I've written that uses *FX calls also illustrates the principles of good procedure writing.

Here are my simple rules:

- Before changing conditions or variables record their state at the beginning of the procedure.
- Reset all conditions and variables at the end of the procedure.
- Make all variables Local.
- No procedure is too trivial to warrant these precautions.

All the above conditions are necessary if procedures are to work under every condition.

Program VI contains a major procedure called PROCartist. It's designed to allow the user to design graphics and text on a Mode 2 screen.

Although it's not complete, it has been designed to be fully expandable to include as many facilities as you wish to put into it.

The procedure has four main parts. The first is a series of osbyte calls that read and record the present status of the function keys *(A%=225)*, cursor keys *(A%=237)* and Escape key *(A%=229)*, and then changes their values.

Next the variables are initialised. *xp%* and *yp%* store the position of the graphics cursor, *plot%* determines the type of graphic output.

Its default value, 5, causes a line to be drawn in the graphics foreground colour. The variables *act%*, *gcol%* and *size%* determine the colour and size of the drawn line.

The repeat loop between lines 280 and 410 receives the keyboard commands and takes the appropriate action. It is here that the program can be developed.

At present the facilities are restricted. f0 to f7 change the colour of the plotted line and f8 and f9 cause flashing to be switched on and off respectively.

The cursor keys cause the graphics cursor to move about the screen. The Escape key causes the procedure to end. All other keys don't yet have a function.

Here are some ideas for developing Program VI:

If S is pressed then the next value input determines the size of the line drawn. Typing T could switch on triangle drawing mode.

The procedure from Program II for drawing circles could be incorporated and called upon if C is pressed.

Obviously once a drawing has been made it ought to be dumped to paper or tape. Those procedures need to be developed.

Program VI illustrates the use of osbyte calls to set up the keyboard for a graphics-based screen editor.

The next example, Program VII, contains similar osbyte calls to set up a text editor.

The overall structures of the two programs are very similar but there are some important differences.

The osbyte calls on lines 160 and 170 change the action of the letter keys when they are pressed at the same time as the Caps Lock/Func key.

The letter A produces character number 240, B character number 241 and so on.

This increases the flexibility of the program as separate functions can be assigned to

```
   10REM Program VI
   20REM
   30REM
   40MODE2
   50PROCartist
   60END
   70DEFPROCartist
   80LOCALoldfn%,oldck%,old
escX,xp%,yp%,gcol%,act%,plo
t%,size%,get%
   90PROCfx(135,0,0)
  100IFY%<>2THENENDPROC
  110PROCfx(225,0,255)
  120oldfn%=x%
  130PROCfx(225,48,0)
  140PROCfx(237,0,255)
  150oldck%=x%
  160PROCfx(237,1,0)
  170PROCfx(229,0,255)
  180oldesc%=x%
  190PROCfx(229,1,0)
  200CLS
  210act%=0
  220xp%=500
  230yp%=500
  240plot%=5
  250gcol%=7
  260size%=48
  270MOVExp%,yp%
  280REPEAT
  290get%=GET
  300IFget%<56ANDget%>47THE
Ngcol%=gcol%ANDBOR(get%-48)
  310IFget%=56THENgcol%=gco
l%ORB
  320IFget%=57THENgcol%=gco
l%AND7
  330IFget%=136THENxp%=xp%-
size%
  340IFget%=137THENxp%=xp%+
size%
  350IFget%=138THENyp%=yp%-
size%
  360IFget%=139THENyp%=yp%+
size%
  370REM insert further fac
ilities
  380REM between these line
s
  390GCOLact%,gcol%
  400PLOTplot%,xp%,yp%
  410UNTILget%=27
  420PROCfx(229,oldesc%,0)
  430PROCfx(237,oldck%,0)
  440PROCfx(225,oldfn%,0)
  450ENDPROC
  460DEFPROCfx(aa%,xx%,yy%)
  470A%=aa%
  480X%=xx%
  490Y%=yy%
  500r%=USR(&FFF4)
  510a%=r%AND&FF
  520x%=(r%AND&FF00)DIV&100
  530y%=(r%AND&FF0000)DIV&1
0000
  540r%=(r%AND&FF000000)DIV
&1000000
  550ENDPROC
```

Program VI

each and every key. For example, line 250 calls PROCa if Func A is pressed.

The program is designed to be expanded to meet your own needs. Procedures that are called when letter and the Func key are pressed should be added to the end of the program.

Let's move on to another area of screen display. It would be wrong for me to discuss the osbyte calls associated with graphics without including *FX20, but first I'll explain the ins and outs of character definitions.

I'm sure that all ardent *Electron User* readers remember our long-running shape dictionary, Casting Agency.

Great ideas were sent in for redefining characters so that they take on the shape of familiar objects.

One example was a crab (August 1984) by Janet Byers of Penrith. Program VIII contains the VDU statements that create the crab. In all seven characters need to be defined. Janet chose characters 232 to 238.

If I wanted to make up further shapes we would choose other numbered characters to modify.

However we may not use more than 32 consecutive characters. This severely restricts the numbers of shapes we can use in any one program.

The solution is to set aside computer memory to store the definitions. This is done with osbyte call *FX20.

The table on page 282 of the Electron User Guide clearly shows the calls that have to be made to reallocate memory for character definition.

For example, if we wish to use the characters 160 to 191 then we have to issue the call *FX20,1.

This makes another 32 characters available for redefinition and the character definition memory is said to be exploded.

There are two major drawbacks to exploding the memory. The computer moves the value of page upwards, and as a result the maximum size of program is reduced.

Secondly, unless your pro-

→

```
   10REM Program VII
   20REM
   30REM
   40MODE6
   50PROCtextwriter
   60END
   70DEFPROCtextwriter
   80PROCfx(135,0,0)
   90IFY%<>6THENENDPROC
  100PROCfx(237,0,255)
  110oldck%=x%
  120PROCfx(237,1,0)
  130PROCfx(229,0,255)
  140oldesc%=x%
  150PROCfx(229,1,0)
  160PROCfx(226,148,0)
  170PROCfx(227,156,0)
  180VDU30
  190REPEAT
  200get%=GET
  210IFget%>135ANDget%<140T
HENVDUget%-128
  220IFget%>31ANDget%<128TH
ENVDUget%
  230IFget%=135THENPROCcopy
  240IFget%=13THENVDU13,10
  250IFget%=240THENPROCfunc
a
  260IFget%=241THENPROCfunc
b
  270IFget%=242THENPROCfunc
c
  280REM extra facilities c
an be entered
  290REM between these two
lines
  300UNTILget%=27
  310PROCfx(229,oldesc%,0)
  320PROCfx(237,oldck%,0)
  330PROCfx(226,1,0)
  340PROCfx(227,1,0)
  350ENDPROC
  360DEFPROCcopy
  370ENDPROC
  380DEFPROCfx(aa%,xx%,yy%)
  390A%=aa%
  400X%=xx%
  410Y%=yy%
  420r%=USR(&FFF4)
  430a%=r%AND&FF
  440x%=(r%AND&FF00)DIV&100
  450y%=(r%AND&FF0000)DIV&1
0000
  460r%=(r%AND&FF000000)DIV
&1000000
  470ENDPROC
```

Program VII

```
   10REM Program VIII
   20REM
   30REM
   40MODE2
   50COLOUR1
   60COLOUR131
   70CLS
   80PRINTTAB(4,28)"Psycho-
Crabs"
   90VDU23,1,0;0;0;0;0;0;0
  100VDU23,232,0,0,0,85,85,
114,34,34
  110VDU23,233,60,2,1,0,7,8
,17,34
  120VDU23,234,62,127,235,2
55,255,221,99,62
  130VDU23,235,30,32,192,12
8,240,136,68,34
  140VDU23,236,4,4,4,0,0,0,
0,0
  150VDU23,237,28,0,0,0,0,0
,0,0
  160VDU23,238,16,16,16,0,0
,0,0,0
  170crab$=CHR$32+CHR$232+C
HR$32+CHR$10+CHR$8+CHR$8+CH
R$8
  180crab$=crab$+CHR$233+CH
R$234+CHR$235+CHR$10+CHR$8+
CHR$8+CHR$8
  190crab$=crab$+CHR$236+CH
R$237+CHR$238
  200REPEAT
  210PRINTTAB(RND(10),RND(2
5));crab$
  220COLOURRND(16)-1
  230inkey%=INKEY(15)
  240UNTILFALSE
```

Program VIII

From Page 45

gram is loaded to the new page value it will be lost.

Bearing those disadvantages in mind, this method still provides an easy way of creating shapes.

Finally this month we'll take a look at the call that returns the VDU status.

*FX117 is used to discover information about scrolling, cursor separation, printing at the graphics cursor and so on. All the data is returned in the X register.

First we'll have to learn a bit about bits!

The number stored in any single memory location or returned by an osbyte call is always between 0 and 255 inclusive. It's not possible to have a larger or smaller number.

In binary the number can be represented by 8 bits which are either 0 or 1. For instance:

decimal	bit number							
	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	1
75	0	1	0	0	1	0	1	1
127	0	1	1	1	1	1	1	1

The first column is worth 128, the second 64, then 32, 16, 8, 4, 2 and finally 1.

For any number that is returned by the osbyte call we can determine whether each individual bit is set at 1 or 0. This then tells us something about the VDU status.

- If bit 0 is 1 then the printer output has been enabled by a VDU2 (or Ctrl B).
- If bit 1 is set then scrolling will not take place.
- If bit 2 is set then paged scrolling has been set by Ctrl N or VDU14.
- If bit 3 is set then a text window has been defined.
- Bit 4 has been reserved for future expansion.
- If bit 5 is set then printing at the graphics cursor is in operation.
- Bit 6 is set when input and output cursors have been separated – that is when the cursor keys are being used to copy.
- Finally bit 7 is set when VDU21 has been used to disable VDU output.

Using PROCfx(117,0,255) and then testing out the information returned in *X*% we can discover the VDU status. Program IX does that for us.

Try setting up a text window or switching the printer driver on (Ctrl B) or using paged mode (Ctrl N) before running the program.

That ends this month's look at osbyte calls. I hope you take up the development of one of the programs.

They've been structured to help programmers extend them to meet individual needs.

Good luck with your endeavours.

```
   10REM Program IX
   20REM
   30REM
   40PROCfx(117,0,255)
   50IFX%AND1THENPRINT"Prin
ter enabled"
   60IFX%AND2THENPRINT"Scro
lling disabled"
   70IFX%AND4THENPRINT"Page
d scrolling set"
   80IFX%AND8THENPRINT"Text
 window defined"
   90IFX%AND32THENPRINT"VDU
5 set"
  100IFX%AND64THENPRINT"Cur
sor keys in use"
  110IFX%AND128THENPRINT"VD
U OUTPUT DISABLED":REM ther
efore this message will not
 appear!
  120END
  130DEFPROCfx(aa%,xx%,yy%)
  140oX=&FFF4
  150AX=aa%
  160XX=xx%
  170YX=yy%
  180rX=USR(oX)
  190=X=(rX AND &FF00)DIV&100
  200ENDPROC
```

Program IX

SLOGGER

ROMBOX

Want to make serious use of your Electron? . . . Then fit the ROMBOX and you will have access to a wide range of ROM based software (including many BBC ROMs).

- ★ Superb value for money.
- ★ Runs all good (non mode 7) BBC ROM software.
- ★ Fully compatible with Plus 1 and Plus 3.
- ★ Up to 8 ROMs instantly selectable.
- ★ Supports 8K and 16K sideways RAM.
- ★ Allows further expansion at rear.

ONLY £44.95

STARWORD (16K ROM)

A Professional word Processor for UNDER £35

★ Does everything VIEW does plus MUCH MORE!

Designed for the home user, education or small business needs, STARWORD enables even those with limited typing skills to produce and print letters, manuals or reports using the Electron.

★ Easy to use.

ONLY £34.50

STARSTORE (ON ROM)

Store and retrieve your names and addresses or any other information with the STARSTORE DATABASE, written specially for the Electron. STARSTORE works with STARWORD for personalising standard letters (mailmerging).

ONLY £29.95

PRINTER DRIVER for STARWORD

Use most printers with STARWORD

Cassette £7.95 ROM £9.95

SLOGGER BUSINESS PACKAGES

ROMBOX-P + STARWORD + STARSTORE + FREE PRINTER ROM.
ONLY £120.00 BP1

ROMBOX + STARWORD + STARSTORE
ONLY £105.00 BP2

TAPE TO DISC FOR CUMANA TP2CU £19.95

STARWORD FOR CUMANA £34.50

Also available:
Vine Micro's ADDCOMM £28.00
Plus 1 ROM upgrade £7.95

Also available:
A wide selection of Rom Software from other leading manufacturers. Please phone for details.

AND NOW . . . ROMBOX-P

Offers all the features of ROMBOX Plus built-in centronics printer interface and FREE Printer ROM (worth over £20).

- ★ Up to 8 ROMs instantly selectable.
- ★ Selectable 8/16K Print buffer to increase throughput.
- ★ Ideal as a word-processing station with Plus 3 fitted.
- ★ FREE Printer ROM included.
- ★ Superb value for money at **ONLY £69.95**

STARMON

- ★ Display of memory in: ASCII and binary, decimal, octal or hexadecimal.
- ★ Full support of sideways ROMs.
- ★ Comprehensive debugging facilities, including breakpoints, traces and events.
- ★ "a very professional piece of firmware" . . . Acorn User.

ONLY £22.50

ELKMAN

The most powerful ROM manager on the BBC is now available for the Electron.

ONLY £17.50

PRINTER ROM

Allows use of sideways RAM as 8K or 16K print buffer and offers a host of useful utilities for EPSON printers. Ideal for all your printing requirements.

★ Electron/BBC compatible.

ONLY £24.95

STARGRAPH

Graphics Rom giving screen dump to EPSON printer, printing of text at any angle or any size, circle, ellipse polygons, arc, dotted lines, colour fitting plus more.

ONLY £21.95

T2P3

Transfer the majority of cassettes Tapes 2 Plus 3 disk system, for much faster loading.

ONLY £19.95

All prices include VAT P&P UK Mainland only

SEND FOR THEM TODAY

Cheques payable to: **SLOGGER LTD.**

☐ Access No. .....

☐ Visa .....

Name .....

Address .....

.....

Signed ..... Tel. No: .....

Send Orders to: **SLOGGER LTD.**
107 RICHMOND ROAD, GILLINGHAM, KENT

PLEASE SEND ME: Cost

1 _____ £ _____
2 _____ £ _____
3 _____ £ _____
4 _____ £ _____
5 _____ £ _____

Total £

DEALER ENQUIRIES WELCOME. TEL: 0634 52303 (2 lines)

Why PROCs give GOSUBs the runa

Let ROGER FROST give your programming technique a powerful booster shot

IN the June issue of *Electron User* we learned about the useful GOSUB command and saw how it enables you to make the program jump to a special section of code dedicated to one particular task.

Subroutines are a powerful facility, and yet whole droves of people despise them fervently and also tend to despise those who use them.

Such people talk of "structure" and will tell you that one of the main advantages of your Electron is its ability to support procedures.

So what's wrong with the dreaded GOSUB? Well, it can be a slow way of going about things. Type in Program I and you'll get the idea.

```
10 REM PROGRAM I
20 MODE 2
30 count=0:TIME=0
40 REPEAT
50 GOSUB 1000
60 UNTIL count=500
70 PRINT'TIME/100
80 REM ADD LINES HERE
990 END
1000 REM COUNTER
1010 count=count+1
1020 RETURN
```

Program I

Here comes the tricky bit, which is to add extra lines 80 to 980. As these are just dummy lines, each contains only a colon (:). This can be done without too much typing by using your Electron's function keys and AUTO.

First set up function key 1 by typing:

```
*KEY1 : M
```

The symbol before M is the control symbol, obtained by pressing the Shift key and the cursor right key. Then enter:

```
AUTO80
```

and if you hold down both the Func and 1 keys the line numbers will appear along with the colon.

You must be prepared to stop at line 980 or you'll overwrite the subroutine at line 1000.

What this marvellous program does is to call the subroutine 500 times and measure how long it takes. On my machine it averages 6.93 seconds.

Why is it so slow when it's not really doing anything?

The answer is because your micro is very stupid and doesn't remember where the subroutine is. Each time it encounters GOSUB 1000 it starts at the first line of the program and says: "Is this line 1000? No!". It then tries each line in turn and continues until finally it finds the one it wants.

If you delete some of the dummy lines (say, 100 to 500), you'll find that the program runs more quickly. This shows that for short programs, subroutines are acceptably fast.

Table I shows how the time taken for subroutines increases as more lines are added to a program.

Now here's a little puzzle for those concerned with program speed. Try renumbering Program I so that it starts at line one and goes up in ones.

```
RENUMBER1,1
```

will do the trick. It's appreciably slower. Perhaps some enterprising person could find out the optimum gap between line numbers and explain why it matters.

Finished pondering that? Let's consider procedures. A procedure (PROC), like a GOSUB, is a body of code dedicated to a single task. However it is very different, not least in that it can be called by name.

When the computer encounters the Basic keyword PROC it jumps to the DEFPROC of the same name. The lines between the DEFPROC and the subsequent ENDPROC make up the body of that procedure.

So if the Electron comes across a PROCFRED it will immediately look for the definition of FRED starting at DEF PROCFRED and perform that code.

To see this in practise, have a look at Program II. It's nearly identical to Program I, but uses a procedure instead of a subroutine.

What happens here is that the computer comes across the keyword PROC at line 50. It immediately looks for the DEFPROC of that name and finds it at line 1000.

It is at this point that we actually define what the procedure called count is to do.

On reaching ENDPROC the definition of the procedure ends and the program control reverts to line 60, the line following the one that called the procedure.

This then sends the micro round the loop again until the final conditions have been met.

END, at line 990, is vital. If you leave it out the program will eventually reach line 1020 and won't know where to go from there. The result is a crash. Try it and see for yourself.

You can think of the procedure as a sort of appendix at the back of a book. When the micro comes across a procedure call it looks up the code in the procedure definition to see what it can do.

When you run this you'll find the same task is done in less time than in Program I. This, of course, can be vital in

```
10 REM PROGRAM II
20 MODE 2
30 count=0:TIME=0
40 REPEAT
50 PROCcount
60 UNTIL count=500
70 PRINT'TIME/100
80 REM ADD LINES HERE
990 END
1000 DEFPROCcount
1010 count=count+1
1020 ENDPROC
```

Program II

Number of lines before subroutine	Time taken (seconds)
10	4.99
20	5.32
40	5.68
60	6.03
80	6.38
100	6.93

Table I: Subroutine dependence on program length

round

an action game.

What's happened is that the computer actually remembers where to find its procedure. So every time it has to PROC-count it can jump straight to it, without having to search through the program as a GOSUB would.

You'll also see from this that procedures are easy to use. You start them with a DEFPROC followed by the title of your procedure and end them with, logically enough, ENDPROC.

Everything between makes up the body of the procedure, the bit that does the work.

To use the procedure you just type PROC and its name. This can be done either in the program or as a command in direct mode.

Table II shows what happens to the running times of Program II with different numbers of dummy lines. You'll see that the time taken is very consistent. Also, except for absurdly short programs, it's faster than subroutines.

Procedures are ideal for those little bits of code that crop up again and again. Also, because you've given them a title, they're easier to understand than subroutines, which can only be given a line number as a label.

This is really helpful when it comes to getting the bugs out of programs. For example, PROCyourturn makes sense – you know that you are looking at a section of code concerned with your turn.

GOSUB 673 may do just the same thing, but it is meaningless and offputting.

If you want to produce listings which can be understood, by other people or yourself, the procedure is far better.

Writing a procedure really means defining a new word for your Electron to understand.

You may want to define a word, which is specific to one program or, even better, you may define a word or procedure for general usage.

For example, if you look through *Electron User*, you'll find procedures for circles or large text which could be incorporated into many of your own masterpieces.

After a while you'll find that you've built up a series of procedures that you use regularly.

Procedures lend themselves to the style of programming known as "top down". At its simplest, in this style your main program is thought out first. It may consist of half a dozen procedures, such as:

```
PROCtitle
PROCsetup
PROCdrawscreen
PROCgame
PROCscore
PROCend
```

This sets out the program structure in outline. It's rather like the contents page of a book, where you can see at a glance what's to come.

After that you move on to producing your actual procedures.

If, like me, you program as a part-time hobby, you can tackle each procedure quite separately and, because the titles mean something, you can return to them weeks later and still make sense of them. Well, in theory you can!

However if you're writing programs over a period of time it's all too easy to get variable names in a muddle.

For instance, you might use the variables *X* and *Y* for the coordinates of a frog, forgetting that you had already used *X* and *Y* for some other purpose in a remote part of the same program. When you are lifting procedures whole from other programs this can easily happen.

This clash of variables could be a disaster, but there is a seemingly impossible trick that can save the day. This allows variables to be two things at once within the same program.

You can "declare your variables local". It's very easy to do, as Program III shows.

In it the first thing that happens is that a couple of variables, *X* and *Y*, are assigned the values 1 and 20. We then move on to the procedure, which defines LOCAL *X* and *Y* variables (line 1010).

Line 1020 gives the variables new values, and the next line displays them. When the procedure finishes program control returns to lines 40 and 50, which print the original values of *X* and *Y*, unaffected by line 1020.

So the computer is able to assign different values to the same variable name in different parts of the program.

Outside the procedure, *X* is 1 and *Y* is 20. Inside it, *X* is 5 and *Y* is 10. Incidentally, variables can be strings of characters just as well as numbers.

What all this means is that you can use your favourite names for variables in lots of places. So long as they're LOCALised, they won't conflict with each other.

It's even more useful to remember LOCAL if you keep a library of standard procedures to build into your programs.

If you've declared the variables used in them as LOCAL you will have avoided the risk of your program crashing as variables get mixed up.

If you tried this with GOSUB all hell would break loose, as the variables in each subroutine are available to the whole program.

So far we've learned that the procedure, with its DEFPROC, PROC and ENDPROC commands, is usually faster than the subroutine, is more meaningful and can have LOCAL variables.

It may beggar belief, but it is possible to use the same procedure code to perform different but related tasks.

This process, known as passing parameters into the procedure, is a really superb feature of Electron Basic, and it is not as painful as it sounds.

To do this, the DEFPROC

```
  10 REM PROGRAM III
  20 X=1:Y=20
  30 PROCdemo
  40 PRINT"Out of procedur
e X= ";X
  50 PRINT"Out of procedur
e Y= ";Y
  60 END
1000 DEFPROCdemo
1010 LOCAL X,Y
1020 X=5:Y=10
1030 PRINT"In procedure X=
";X
1040 PRINT"In procedure Y=
";Y
1050 ENDPROC
```

Program III

Number of lines before procedure	Time taken (seconds)
10	5.05
20	5.05
40	5.05
60	5.05
80	5.06
100	5.06

Table II: Procedure independence of program length

Nightingale
The Modem

FREE
ENTRY COMPETITION
6 NIGHTS IN NEW YORK

The facts speak for themselves, **Nightingale** is without equal in its field. It is undoubtedly the most popular multi baud rate modem available today. Whatever your communication requirements, the simplicity of use and the versatility of **Nightingale** will enable you to provide a solution whether the need is to extract information from Prestel, send telex's via Telecom Gold, download telesoftware or exchange files, **Nightingale** will do the job and a whole lot more besides.

Nightingale operates at the two most popular baud rates 1200/75 as used by both private and public viewdata systems and 300/300 baud for communication with remote mainframe, mini or micro systems. Also available in 'reverse viewdata mode' 75/1200.

Nightingale has passed the very rigorous tests required by the BABT and is fully approved for use in the UK. This not only means that you can rely upon the safety of the product because of the stringent manufacturing constraints, but also means that because it is a BABT approved product, you can be assured that your application to Telecom Gold, Prestel etc., will not be refused.

APPROVED for use with telecommunication systems run by British Telecommunications in accordance with the conditions in the instructions for use

Tellstar *The Software*

A fitting partner for the Nightingale modem, Tellstar has been developed from the same stable, and to the same high specification as the best selling BBC communications software COMMSTAR. With its powerful combination of Viewdata and ASCII terminal capabilities contained in one Eprom, Tellstar is the first real communications software available for the Electron and probably the only communications software you will ever need. It will allow you to use to the full, the versatility of the Nightingale modem in accessing information from Prestel, sending/receiving files or sending telex's, downloading telesoftware, etc. Although it is unusually versatile, it is extremely easy to use. A comprehensive manual describes each of Tellstars features in simple terms. Tellstar comes complete with an RS423 Interface developed to an Acorn specification and manufactured by Pace which is connected to your Electron via the Plus 1 Interface.

The Competition

Pace have linked with the premier computer information provider on Prestel, MICRONET and Virgin Atlantic Airlines, all pioneers in their own fields, to offer you a challenge. A chance to try out your skills using your modem and play the 'Pace round Britain Quiz' on Micronet, and in so doing, an opportunity to win a fabulous holiday for two in New York.

This competition will be open to you between November 1st and February 1st if you buy one of our modems before the end of January. In addition, you will be given, free of charge, a full quarters subscription to MICRONET worth £10, and free registration to Microlink on Telecom Gold.

The prize includes a return flight to New York, 6 nights accommodation in a high quality New York Hotel, and spending money to the total value of £400.

Full details are contained with each modem.

micronet 800

Virgin atlantic

Combination of Nightingale Modem, Tellstar and RS423 Interface, special introductory offer ~~£139~~ £119 plus VAT

PACE

The products and information are available from good quality dealers throughout the country or direct from Pace. Payment accepted by cheque, Access or Visa.

Pace Micro Technology Ltd.,
Juniper View, Allerton Road, Bradford BD15 7AG.
Tel. 0274 488211 Telex 51564
Telecom Gold mailbox No. PCE 001 Prestel Mailbox No. 274729306

From Page 49

command has some special variables or parameters put in brackets after the name. These variables can be real, integer or string or any combination of them. The procedure definition takes the general form:

```
DEFPROCname(A,B,C$)
```

The variables can have any name you care to give them. If you make them meaningful your program will be more understandable. Examples could be:

```
DEFPROCarea(length,width)
```

or

```
DEFPROCcar(make$,model$,yea
r)
```

Whatever your DEFPROC, it will be followed by the section of code that makes up the procedure and terminated with ENDPROC.

The PROCname example above could contain code to print people's names at different positions on the screen.

Within the procedure, *A* and *B* could be the coordinates of the point on the screen where you want the name *C$* to be printed.

To use this procedure, you call it with PROCname followed by the values you wish to assign to the parameters. In your example these are *A*, *B* and *C$*.

A possibility would be to call:

```
PROCname(5,6,"Fred")
```

which would print Fred at screen position 5,6. At another point in the program you could call:

```
PROCname(1,2,"Jill")
```

to print Jill at position 1,2. Although the procedure has done two different jobs, the same piece of code is used, only the parameters have changed.

The procedure parameters are different from other variables. No value has to be assigned to them. In fact, they have no value until they receive one from the procedure call.

A big plus for these parameters is that they are LOCAL to the procedure without having to be declared LOCAL. In our PROCname example, *A*, *B* and *C$* could be used elsewhere in the program with no problem.

There could be occasions when you didn't know what value you wanted to pass to a procedure.

You might, for example, want it to depend on something INPUT while the program was running or on the result of an IF . . . THEN statement. No problem!

Your procedure call can be followed by variable names rather than by actual values. This could look like:

```
PROCname(xposition,ypositio
n,name$)
```

These variables would need values assigned to them during a program. Names might be read from DATA lines to give values to *name$*, and *xposition* and *yposition* could depend on the length of the name. When the call:

```
PROCname(xposition,ypositio
n,name$),
```

is made these values are transferred to *A*, *B* and *C$* for the procedure to use as dictated in the definition.

Then *exposition*, *yposition* and *name$* would take no part in the procedure.

You can look on the variables used in the procedure definition as dummies. They show the Electron what information to expect from the program when the procedure is called and also what to do with it.

So you could have:

```
DEFPROCname(A,B,C$)
```

which uses the variables *A*, *B* *C$* as dummies to show the micro what to do. Then when the procedure is called with actual values, such as:

```
PROCname(5,6,"FRED")
```

or with variables holding values:

```
PROCname(xposition,ypositio
n,name$)
```

the micro uses these values as shown by the way the dummies are used in the procedure definition.

It's a bit like the rule for the area of a rectangle, width times breadth. This is the rule, but if someone told you to find the area of a rectangle with length 5 and height 10, you'd still use the rule.

You'd just substitute length and height for width and breadth. The names may be different, but the rule's the same.

Similarly, the names used for parameters in a procedure call may differ from those used in the procedure definition, but provided they're given in the same order the Electron can get on with the job.

Remember that by passing parameters your procedures can perform variations on a theme.

Supposing you wanted to draw a block of flats. You might need one big square for the walls and lots of smaller squares for the windows:

The procedure in Program IV (lines 1000-1060) which has four number parameters passed to it could perform the trick.

The parameters *x* and *y* refer to the bottom left hand corner of the square, *s* gives the length of each side and *c* the colour of the square.

Of course these parameters have no value until they are passed by the procedure calls, lines 30-50. These won't actually draw a block of flats. That's a task for you to have a go at, but Program IV should point you in the right direction.

All you need to do to get that block of flats is to call the procedure with the correct parameters – PROCsquare with four numbers.

If you could write simple procedures for triangles and circles – or borrow them from other listings – you'd have basis for a design program.

It's all amazingly easy and it really is worthwhile building up a library of useful procedures which can be added to your program.

Although the parameters which you pass to a procedure can be real numbers, integers or strings, you must make the types of variables after the PROC command match those after the DEFPROC.

You can't try to pass a string if the procedure is expecting a number or vice versa. This may seem obvious, but it's amazing how many times it can crop up.

You can, however, mix real and integer numbers, but remember that integers are always just whole numbers with no fraction part.

Finally, you must use the correct number of parameters. If you were to change line 30 in Program IV to:

```
30 PROCsquare(100,200,300)
```

the program would crash because the DEFPROC square command is expecting to receive four parameters and you are only giving it three.

Don't worry if this sounds a little complex, with practise it'll all become easy.

So, powerful though the subroutine is, you'll see that the procedure has several advantages.

For starters it's usually faster, as we saw in the tables. Also it allows you to pass parameters and make use of the handy LOCAL command.

By adopting a programming style based on procedures your work will look straightforward and meaningful.

This structure will not only please all those who read your listings, but should help you to develop a sensible, problem-solving approach to programming.

It's easy to use a GOSUB when, for a little more time and effort, you could use procedures. However don't be tempted, use procedures for preference.

And when you do you'll find your programming becomes more powerful and more pleasurable.

```
   10 REM PROGRAM IV
   20 MODE2
   30 PROCsquare(100,200,30
0,4)
   40 PROCsquare(300,200,45
0,2)
   50 PROCsquare(425,570,40
0,1)
   60 END
 1000 DEFPROCsquare(x,y,s,c
)
 1010 GCOL0,c
 1020 MOVE x,y
 1030 MOVE x+s,y
 1040 PLOT85,x,y+s
 1050 PLOT85,x+s,y+s
 1060 ENDPROC
```

Program IV

AMSTRAD £8.95

ZX SPECTRUM £7.95

COMMODORE 64 £8.95

Yie Ar KUNG-FU

Imagine Software is available from: Boots, WHSMITH, John Menzies, WOOLWORTH, LASKYS, Rumbelows, Greens, Spectrum Shops and all good dealers.

Imagine Software (1984) Limited • 6 Central Street • Manchester M2 5NS • Tel: 061 834 3939 • Telex: 669977

MEKON RAIDERS

NEW
For ELECTRON & BBC

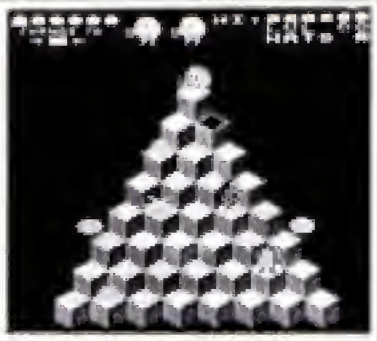
A TACTICAL ARCADE ACTION GAME

- **3D graphics**
- **Tactical grid**
- Defend earth from the advancing Mekon raiders.
- Battle through space attacks.
- Descend to the surface of the Mekon battle stars.
- Destroy the Mekon battlestars before they destroy earth.
- Exciting – fast machine code action.

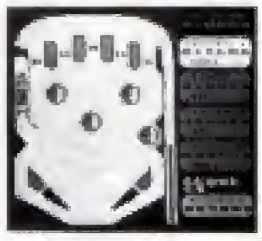
£6.95 INC VAT & POSTAGE

ER*BERT

Join ER*BERT in his zany cubic domain. Avoid coily – Boris – and other unwelcome guests. Four different scenes each with 10 levels, fast – funny, addictive!

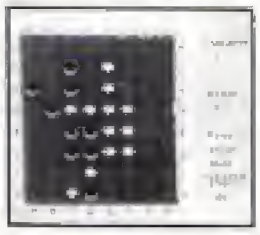


£4.95 INC VAT & POSTAGE



PINBALL & REVERSI

Twin pack of competitive games – arcade action – strategy and skill.



£5.95 INC VAT & POSTAGE

Games are available for Electron & BBC by mail order from Microbyte Software Post FREE.

Microbyte Software (EL 12)
18 Hilgrove Road
Newquay
Cornwall TR7 2QZ.

ACCESS HOTLINE 24 HR.
06373 6886

MICROBYTE SOFTWARE

FACT FILE

500

for BBC and Electron

Expand your KOSMOS ANSWER BACK QUIZ with

NATURAL HISTORY	(Ages 10+)	£3.95 ■
ARITHMETIC	(Ages 6-11)	£3.95 ■
SPELLING	(Ages 8-12)	£3.95 ■
ENGLISH WORDS	(Ages 12+)	£3.95 ■
KNOW ENGLAND	(Ages 12+)	£3.95 ■
KNOW SCOTLAND	(Ages 12+)	£3.95 ■
FIRST AID	(Ages 13+)	£3.95 ■
GENERAL SCIENCE	(Ages 14+)	£3.95 ■
JUNIOR GENERAL KNOWLEDGE *	(Ages 6-11)	£3.95 ■
SENIOR GENERAL KNOWLEDGE *	(Ages 12+)	£3.95 ■
SPORT *	(Ages 14+)	£3.95 ■

** Same content as ANSWER BACK program of same title*

All FACTFILE 500 packs are compatible with
ANSWER BACK "JUNIOR" QUIZ "SENIOR" QUIZ and "SPORTS" QUIZ
YOU MUST HAVE AN "ANSWER BACK" PROGRAM TO USE THE FACTFILES

Each FACTFILE 500 pack contains an amazing database of 500 fascinating questions and 2000 multiple choice answers. The above list is only the start, dozens more leisure and educational titles will be added to the range shortly

All FACTFILES are also available on 40 or 80 track BBC disc, £5.45 each

ALL PACKS INCLUDE £50 PRIZE DRAW ENTRY COUPON

ALL PRICES include VAT, P & P and 24-hour despatch
Send cheque, PO or quote your Access number

WIZARD SOFTWARE

1 Pilgrims Close, Harlington, DUNSTABLE, Beds. LU5 6LX
Tel. (05255) 3942

LOTHLORIEN

NEW

Waterloo

- 1 or 2 player all graphic wargame
- Keyboard control
- Simultaneous movement by both sides
- Artillery bombardment with target selection
- 10 levels of play
- Comprehensive booklet
- Cassette £9.95 BBC or Electron

ALSO AVAILABLE FOR BBC AND ELECTRON:
Special Operations £6.95 Redcoats £6.95 Paras £6.95

THE WARMASTER CLUB – Join Today

- Free Membership
- Special Offers
- FREE Lothlorien Game if you join on this coupon AND buy Special Operations.
- Free Newsletter
- Special Club Editions
- Advance Information on new Lothlorien Wargames

M C Lothlorien Ltd, 56A Park Lane, Poynton Tel: 0625 876642

Please send me _____

Computer _____ Please enrole me in the Warmaster Club ☐

I enclose a cheque/P.O. made payable to M C Lothlorien Ltd
Please debit my Access Account No./VISA

Signature _____ Name _____

Address _____

_____ Post Code _____

M C LOTHLORIEN, 56a Park Lane, Poynton, Cheshire SK12 1RE. Tel: Poynton (0625) 876642

REF EU/2

IMPORTANT NOTICE TO ALL OWNERS OF ELECTRON COMPUTERS

If you want the best in software for your machine in the future make sure you fill in the Registration Form below to register on our Mail List to be regularly informed of our latest releases. Our first batch of titles will include Jet Set Willy and Ian Botham's
Test Match.

Send your completed form now to:

Tynesoft,
Freepost NT 873,
Unit 3,
Addison Industrial Estate
Blaydon-on-Tyne NE21 4BR
(No stamp required)

Title (tick) Mr Mrs Miss

Initials

Surname

Address

Electron

Post code

Telephone no. (including STD code)

Now YOU – and your Electron – can enter the exciting, ever-changing world of telecomputing!

The fastest growing, most exciting field of micro-computing is communications. Just one phone call away there's a vast range of fascinating electronic information and services such as:

★ ***MicroLink:*** *A specialised service on Telecom Gold, Britain's leading carrier of electronic mail. MicroLink gives you your own private letter box, the ability to send and receive telex messages, to send telemessages for next-day delivery to any address in the UK or USA, ready access to up-to-the-minute computer news, a large range of mainframe services and much, much more.*

★ ***Prestel/Micronet:*** *More than 300,000 pages of information, electronic magazines and special interest clubs, with news, advice and reviews. And there's a wealth of software just waiting to be downloaded directly into your Electron.*

★ ***Bulletin Boards:*** *News, views, software and specialist help – they're all available on the scores of home-brewed electronic notice boards that are springing up all over the country.*

For many months we at *Electron User* have been looking for a communications package sophisticated enough to meet our exacting requirements.

Now at last it's here – the Tellstar/Nightingale combination.

The Pace Nightingale is a tried and tested, fully BT approved, multi-baud modem that plugs into a standard telephone socket.

Telstar complements it perfectly. It allows easy access to all the major electronic information providers, including MicroLink and Prestel, and while not available in full colour it contains all the other features expected of advanced communications software.

And of course, the package includes the necessary serial interface to link directly with your Electron via your Plus 1.

Now there's nothing to stop you playing your part in the communications revolution – dial up, log on and have fun!

This *Electron User* communications package includes the Pace Nightingale modem, RS423 interface and Tellstar software, FREE registration to MicroLink (worth £5), FREE quarter's subscription to Micronet (worth £10), and your chance to win a fabulous holiday in New York! *(The Plus 1 is not included.)*

Normal retail price: £160.95 (incl. VAT)

Electron User special price: £145 **(SAVE £15.94)**

Including VAT, postage & packing

TO ORDER, PLEASE USE THE FORM ON PAGE 61

If you want to start doing more with your micro than just playing games, this package is your ideal introduction to the four most popular applications for professional computers. All the programs have been designed for simplicity, so even a child can use them. Yet they include advanced features not yet available on programs costing many times as much!

Finalist for the Home Software of the Year Award

BRITISH MICRO COMPUTING AWARDS 1985

MINI OFFICE

Contains 32 page booklet giving clear, easy-to-follow instructions on all 4 programs

Word Processor: Ideal for writing letters and reports. There is a constant display of both time and word count, plus a words-per-minute display to encourage the budding typist! A unique feature is the double-size text option in both edit and printer mode – perfect for young children and people with poor vision.

Database: You use this for storing information, just like an office filing cabinet. Facts you have entered can be quickly retrieved by just keying in a word or part of a word. They can be sorted, replaced, saved for future use or printed out.

Spreadsheet: Enables you to use your micro for home accounts or pocket money records. It creates a display of numbers in rows and columns. Continuous updating is possible, and a changed figure can be instantly reflected throughout the rest of the spreadsheet. Your results can be saved, to be used for future updates, or can be fed into its associated program . .

Graphics: Part of the spreadsheet section, it lets you draw bar charts, pie charts and histograms to give a graphic presentation of your statistics. Helps to give life and colour to the dullest figures!

☆ Word Processor ☆ Spreadsheet
☆ Database ☆ Graphics

Now they're all together – in ONE simple package

Four full-scale programs for an incredible low, low price! **£5.95** cassette

Also available from:
BOOTS COMET Currys Dixons WHSMITH
Greens John Menzies LASKYS Spectrum *and other computer stores*

Use the special order form on Page 61 – No stamp needed!

All programs classroom tested and educationally approved

FUN SCHOOL!

10 programs on each cassette

Early learning – the easy way on the BBC Micro and Electron

Help with reading:

From simple shapes and letters to spelling and anagrams

3 cassettes for 3 age groups

- Under-5's
- Ages 5-8
- Ages 8-12

Help with maths:

From number recognition and counting to simple sums and games of logic

Only £4.95 each cassette!

Use the special order form on Page 61. No stamp needed.

Listings galore!

ONLY £3.75 each

Save yourself the chore of typing in listings by sending for our monthly tapes, packed with games, utilities, graphics and other programs from the pages of *Electron User*.

On the January 1986 tape:
FRUIT WORM An arcade classic. **HELICOPTER RESCUE** Pilot an air sea rescue helicopter. **MACHINE CODE** Detect collisions between sprites. **TAPEDISC** Transfer your software to disc. **MODE012** Multi-Mode screens. **TRICIRC** A circle of triangles.

On the December 1985 tape:
GET SET SANTA Christmas fun collecting presents. **MISSILE ATTACK** Save your cities! **PROGRAM PROBE** Using joysticks. **SPACE COUNT** Counting for youngsters. **CHRISTMAS CARD** Cards and carols for all. **DISC MENU** Disc Menu creator. **NOTEST** Music tutorial. **NOTEBOOK** Bubble sort.

On the November 1985 tape:
KARATE WARRIOR Electrifying combat. **ULA** Mode 6 Mode 7! **PAINT ROLLER** Colourful arcade action. **DEFUSE** Beware the bombs. **SPRITE PRINT** Machine code graphics utility. **TRAIN** Far from stationery graphics. **TABLE MOUNTAIN** Sums and summits.

On the October 1985 tape:
DUNGEON QUEST An amazing all action arcade adventure. **PILOT** Computer assisted learning language. **RAVING ROLLER** Arcade action in the garden. **TRAIN** Animated action. **KALEIDOSCOPE** Colourful graphics action.

On the September 1985 tape:
TEXNDAN 3D Wild West shootout. **PINTCURSOR** Machine code graphics. **SPRITE/ED** Sprite editor. **COMPOSE** Writing music simplified. **REVERSI** Clensing strategy game. **SIMPLEFILE** Save and read data. **BOUNCE BALL** Two player action. **ROTATE** Animation in a spin.

On the August 1985 tape:
DIGGA Exciting arcade action beneath the earth. **DODGE THE ASTEROIDS** Fun deep in space among the asteroids. **M/CODE GRAPHICS** Sliding pints of beer! ***FX** The OS explored. **MOVEIT** An intriguing sliding puzzle. **HEXGRAM** An educational game to increase your word power.

On the July 1985 tape:
MANIC MOLE Machine code action at its best. **HIGHER OR LOWER** Guess the card. **TIME BOMB** Carefully collect TNT. **M/CODE GRAPHICS** Two demonstrations. **FX1/2** The OS on call. **PIRATE MATHS** Sum fun. **NOTEBOOK** Password Generator.

On the June 1985 tape:
QUASIMODO Bellringing classic. **DISASSEMBLER** Machine code utility. **ACTIVITIES** Educational fun. **REFLECT** Aggressive aliens. **ENGINE** Animation. **DODGE** Race track action. **STRINGALONG** Scrolling fun. **CASTLE** Medieval graphics. **MATHS CURVE** Angles and art. **NOTEBOOK** Trees.

On the May 1985 tape:
SKRAMBLE! Compulsive arcade action. **SHEEPNIM** The logic game. **TEXTWRITER** Screen utility. **LIFE** A cultured classic. **CEDRIC** Educational fun. **THREE-D** Outstanding utility. **SPOKES** Fascinating graphics. **MOONORBIT** Heavenly displays. **BLAZON** Heraldic devices. **FLOWERS** A Basic bouquet. **NOTEBOOK** Annotated alphabet.

On the April 1985 tape:
SUPER ARCHER Target practice. **BINARY SEARCH** Search data efficiently. **JOYPLUS** Switched joystick routine. **ODD ONE OUT** Educational fun. **POLYGONS** 3D rotation. **MONEY CRAZY** Arcade action. **STARCHART** The night sky. **FORTUNE TELLER** Horoscope. **COLLISION DETECTION** Alien encounters. **HILO** Guessing game. **NOTEBOOK** Hello to assembler.

On the March 1985 tape:
MR. FREEZE Ice cube arcade action. **SCREENDUMP** Two procedures for printer dumps. **FILLER** The machine code fill routine. **FRED'S WORD GAME** Educational fun. **BIG LETTERS** Large text utility. **PERCY** Beat the burning fuse. **ANIMATION** Two example programs. **PIGS** Flying bacon. **NOTEBOOK** Display formatting.

On the February 1985 tape:
CRAAL The mystifying maze adventure. **BOUNCY** Addictively annoying action. **PAIRS** Can you remember the cards? **BASE** A Binary/hexadecimal conversion utility. **CATCHER** Collect the eggs before they break. **CLOCK** Time keeping utility. **RACER** Grand Prix action. **NOTEBOOK** Graphics windows. **TRIG** All the right angles.

On the January 1985 tape:
SPACE BATTLE Destroy the deadly descending aliens! **NEW YEAR** A sound and graphics greeting. **ESCAPE FROM SCARGOV** Minefield action. **PIE CHART** Statistics made simple. **CLAYPIGEON** An Electron birdshoot. **ORGAN** Music maestro please! **NOTEBOOK** An original program. **RANDOM NUMBERS** Or not so random? **SNAKES** Reptilian arcade action. **CHEESE RACE** Beat rival mice.

On the December 1984 tape:
CHRISTMAS BOX Align the presents logically. **SILLY SANTA** Sort out the muddle. **SNAP** Match the Xmas pictures. **RECOVERY** The Bad Program message tamed. **CAROL** Interrupt driven music. **AUTODATA** A program that grows and grows. **NOTEBOOK** Simple string handling.

On the November 1984 tape:
STAR FIGHTER Anti alien missions. **SCROLLER** Wrap around machine code. **URBAN SPRAWL** Environmental action game. **SPELL** Alphabetic education. **JUMPER** Level headed action. **CAESAR** Code breaking broken. **KEYBOARD** Typing game.

On the October 1984 tape:
BREAKFREE Classic arcade action. **ALPHASWAP** A logic game to strain your brain. **SOUND GENERATOR** Tame the Electron's sound channels. **MULTICHARACTER GENERATOR** Complex characters made simple. **RIGEL 5** Out of this world graphics. **MAYDAY** Help with your morse code. **NOTEBOOK** Palindromes and string handling.

On the September 1984 tape:
HAUNTED HOUSE Arcadle action in the spirit world. **SPLASH** A logic game for non-swimmers. **SORT SHOWS** How sorting algorithms work. **SORT TIME** The time they take. **CLASSROOM INVADERS** Multicoloured characters go to school. **SAILOR** Nautical antics. **MATHS TEST** Try out your mental powers.

On the August 1984 tape:
SANDCASTLE The Electron seaside outing. **KNOCKOUT** Bouncing balls batter brick walls. **PARACHUTE** Keep the skydivers dry. **LETTERS** Large letters for your screen. **SUPER-SPELL** Test your spelling. **ON YOUR BIKE** Pedal power comes to your Electron. **SCROLLER** Sliced strings slide sideways.

On the July 1984 tape:
GOLF A day on the links with your Electron. **SOLITAIRE** The classic solo logic game. **TALL LETTERS** Large characters made simple. **BANK ACCOUNT** Keep track of your money. **CHARTIST** 3D graphs. **FORMULAE** Areas, volumes and angles.

On the June 1984 tape:
MONEY MAZE Avoid the ghosts to get the cash. **CODE BREAKER** A mastermind is needed to crack the code. **ALIEN** See little green men on the Electron way! **SETUP** Colour commands without tears. **CRYSTALS** Beautiful graphics. **LASER SHOOT OUT** An intergalactic shooting gallery. **SMILER** Have a nice day!

On the May 1984 tape:
RALLY DRIVER High speed car control. **SPACE PODS** Mom aliens to annihilate. **CODER** Secret messages made simple. **FRUIT MACHINE** Spin the wheels to win. **CHASER** Avoid your opponent to survive. **TIC-TAC-TOE** Electron noughts and crosses. **ELECTRON DRAUGHTSMAN** Create and save Electron masterpieces.

On the April 1984 tape:
SPACEHIKE A hopping arcade classic. **FRIEZE** Electron wallpaper. **PELICAN** Cross roads safely. **CHESSTIMER** Clock your moves. **ASTEROID** Space is a minefield. **LIMERICK** Automatic rhymes. **ROMAN** Numbers in the ancient way. **BUNNYBLITZ** The Easter program. **DOGDUCK** The classic logic game.

On the March 1984 tape:
CHICKEN Let dangerous drivers test your nerve. **COFFEE** A tantalising word game from Down Under. **PARKY'S PERIL** Parky's lost in an invisible maze. **REACTION TIMER** How fast are you ? **BRAINTEASER** A puzzling program. **COUNTER** Mental arithmetic can be fun! **PAPER, SCISSORS, STONE** Out-guess your Electron. **CHARACTER GENERATOR** Create shapes with this utility.

On the February 1984 tape:
NUMBER BALANCE Test your powers of mental arithmetic. **CALCULATOR** Make your Electron a calculator. **DOILIES** Multi-coloured patterns galore. **TOWERS OF HANOI** The age old puzzle. **LUNAR LANDER** Test your skill as an astronaut. **POSITRON INVADERS** A version of the old arcade favourite.

On the introductory tape:
ANAGRAM Sort out the jumbled letters. **DOODLE** Multicoloured graphics. **EUROMAP** Test your geography. **KALEIDOSCOPE** Electron graphics run riot. **CAPITALS** New upper case letters. **ROCKET, WHEEL, CANDLE** Three fireworks programs. **BOMBER** Drop the bombs before you crash. **DUCK** Simple animation. **METEORS** Collisions in space.

See order form on Page 61

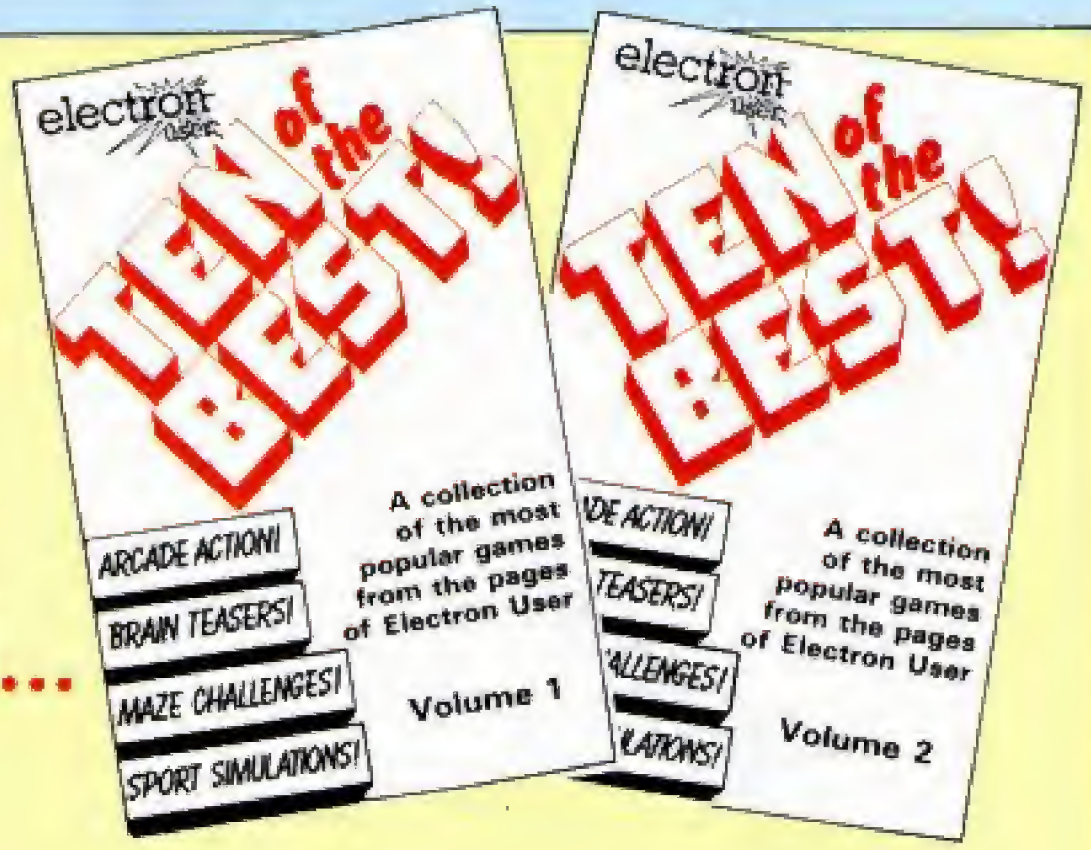
WORN OUT with wordprocessing?

DEPRESSED with databases?

OPPRESSED with machine code?

Then you need...

(It's the perfect antidote to microcomputer malaise!)



These two cassettes are not only crammed with 18 of the best games from the early days of Electron User. At no extra expense, we've included on each an unpublished Roland Waddilove machine code masterpiece as a freebie. Roland's Jam Butty and Atom Smash are arcade action at its fastest and most frustrating. And they're only available with Ten of the Best. So give yourself a treat . . . with the most popular cassettes we've ever produced.

ONLY £5.95 each

Volume 1 contains:

Jam Butty
Machine code simulation of high drama on a building site
Golf
Play a round by yourself, or play against your pals.
Haunted House
Fight against all the odds to get out alive.
Space Hike
Another classic. Help the spacemen avoid maurading monsters.
Parky's Peril
Help Parky through an invisible maze, racing against time.
Rally Driver
All the thrills of high-speed driving, with none of the risks.
Alphaswap
Your letters are in a twist. Can you put them in order?
Knockout
Fast and furious action as you batter down a brick wall.
Money Maze
Avoid ghosts and collect coins in an all-action arcade classic.
Lunar Lander
The traditional computer game specially written for the Electron.

Volume 2 contains:

Atom Smash
Machine code thrills as you help to save the world from destruction.
Bunny Blitz
Go egg collecting, but keep away from the proliferating rabbits.
Castles of Sand
Build castles – but beware the rising tide and hungry sandworms.
Reaction Timer
Test your reactions with this traffic lights simulation.
Solitaire
The Electron version of the age-old game of logic and patience.
Jumper
Jump for your life in this exciting arcade action game.
Break free
Test your wits and reflexes in this popular classic ball game.
Code Breaker
Crack the code in a colourful if frustrating brainteaser.
Parachute
Save the plunging sky divers from a watery end.
Star Fighter
Attack the bandit ships in this fast-moving 3D punch-up.

TO ORDER, PLEASE USE THE FORM ON PAGE 61

EXCLUSIVE!

SAVE £5.95

2 bargain offers for subscribers!

Be one of the first to play Imagine Software's superb simulation of the classic martial art – Yie Ar Kung Fu!

Setting new standards in graphics animation, the high-speed action involves defeating EIGHT successively more aggressive opponents – all hell-bent on re-arranging various parts of your anatomy!

Yie Ar Kung Fu has just hit the shops – at a retail price of £8.95. But we'll send it to you for just £3 when you take out a subscription on the form opposite.
(Not available without subscription.)

£3

To defend yourself you've a choice of manoeuvres that read like an excerpt from the St. Trinians' etiquette book: Flying kick, flying punch, roundhouse kick, lunge punch . . . There are ten in all, all guaranteed to bring life back into your run-of-the-mill Saturday night brawl.

So, Grasshopper, if you want to know the inner secrets of the martial arts, but lack the energy needed for the Dojo – Yie Ar Kung Fu is what you're after.

If you already have a subscription which does not expire for several months you can re-order early to benefit from both these exceptional offers.

SAVE £2

Here's a New Year gift that will serve you well during the whole of 1986!

This 138 page pocket reference diary, published by Pitman contains all the key facts about programming the Electron in a handy, easy-to-use form.

In addition to the one-week-per-page diary, its reference section features a concise guide to the Electron's Operating System, gives the 6502 instruction set and a summary of Basic keywords, and presents vital data about sound and graphics.

The Acorn Reference Diary is now on sale in the shops for £3.95. But we'll send it to you for just £1.95 when you take out a subscription on the form opposite.
(Not available without subscription.)

£1.95

electron user MAIL ORDER OFFERS electron user ORDER FORM

All prices include postage, packing and VAT

Overseas orders sent exclusively by Air Mail

Valid to January 31

Please enter number required in box — £ p

Electron User tapes

£3.75 UK
£4.75 Europe/ Overseas

26 introductory programs		3032
Lunar Lander	1984	3033
Chicken	1984	3034
Spacehike	April 1984	3035
Rally Driver	May 1984	3036
Money Maze	June 1984	3037
Golf	July 1984	3039
Castles of Sand	Aug 1984	3039
Haunted House	Sept 1984	3040
Breakfree	Oct 1984	3041
Star Fighter	Nov 1984	3042
Christmas Box	Dec 1984	3043
Space Battle	Jan 1985	3044
The Kingdom of Craal	Feb 1985	3045
Mr Freeze	Mar 1985	3046
Super Archer	April 1985	3047
Skramble	May 1985	3048
Quasimodo	June 1985	3049
Manic Mole	July 1985	3050
Digga	Aug 1985	3051
Tex'n' Dan	Sept 1985	3052
Dungeon Quest	Oct 1985	3053
Karate Warrior	Nov 1985	3054
Get Set Santa	Dec 1985	3055
Fruit Worm	Jan 1986	3056

Electron User annual subscription

UK & Eire (Sterling only)	£12	3001
Europe	£20	3003
Overseas	£35	3004
Yie Ar Kung Fu	*£3.00	3086
Acorn Reference Diary	*£1.95	2367

***Only available if accompanied by subscription order.**

Commence with _____ issue

Cassette tape annual subscription

£35 UK
£45 Europe/Overseas — 3005

Commence with _____

>>>

Electron User back issues

£1.50 UK
£1.75 Europe
£2.25 Overseas

Set of 4 introductory issues £2	3006
February 1984	3007
March 1984	3008
April 1984	3009
May 1984	3010
June 1984	3011
July 1984	3012
August 1984	3013
September 1984	3014
October 1984	3015
November 1984	3016
December 1984	3017
January 1985	3018
February 1985	3019
March 1985	3020
April 1985	3021
May 1985	3022
June 1985	3023
July 1985	3024
August 1985	3025
September 1985	3026
October 1985	3027
November 1985	3029
December 1986	3029

Education Specials

Classroom Computing on the Electron

Vol. 1

Electron Cassette	£5.95	3070
Micro User Education Special No. 1	£1.95	2210 N/A

Vol. 2

Electron Cassette	£4.95	3076
Micro User Education Special No. 2	£1.50	2320
Both the above	£6.00	3077

Europe/Overseas – add £1 per product ordered

Fun School tape

£4.95 UK
£5.95 Europe/Overseas

Under-5s	3080
Ages 5-8	3081
Ages 8-12	3082

Mini Office

£5.95 UK
£6.95 Europe/Overseas — 3062

Classic Arcade Games

£5.95 UK
£6.95 Overseas — 3064

Ten of the Best

£5.95 UK
£6.95 Europe/Overseas

Volume 1	3068
Volume 2	3069

Electron Plus 1

£39.95 UK — 3084

Electron Workstation Package

£69.95 UK — 3085

Electron Language Lab

£89.95 UK — 3087

Electron Data Manager Package

£149.95 UK — 3086

UK only

Getting Started in BBC Basic

£5.95 UK
£7.45 Europe
£8.45 Overseas — 2289

Communications Pack

£145 UK
£152 Overseas — 3083

>>>

Dust cover

£3.95 UK
£4.95 Europe/Overseas — 3058

Binder

£3.95 UK
£6.95 Europe/Overseas — 3059

Now please complete panels overleaf **TOTAL** _____

EU1

Keep your collection of *Electron User* complete with these handsome binders

Bound in attractive red pvc with the *Electron User* logo in gold blocking on the spine, this binder will hold 12 magazines firmly secured in place by metal rods. **£3.95**

Your Electron needs protecting

Protect your Electron with our luxury dust cover made of soft pliable water-resistant vinyl, bound with strong cotton and decorated with the *Electron User* logo. **£3.95**

You can also order by phone

061-480 0171

Payment: please indicate method (✓)

☐ *Access/Mastercharge/Eurocard*

☐ *Barclaycard/Visa*

☐ *Cheque/PO made payable to Database Publications Ltd.*

BLOCK CAPITALS PLEASE

Name _____

Address _____

_____

_____

_____ Post Code _____

Signed _____

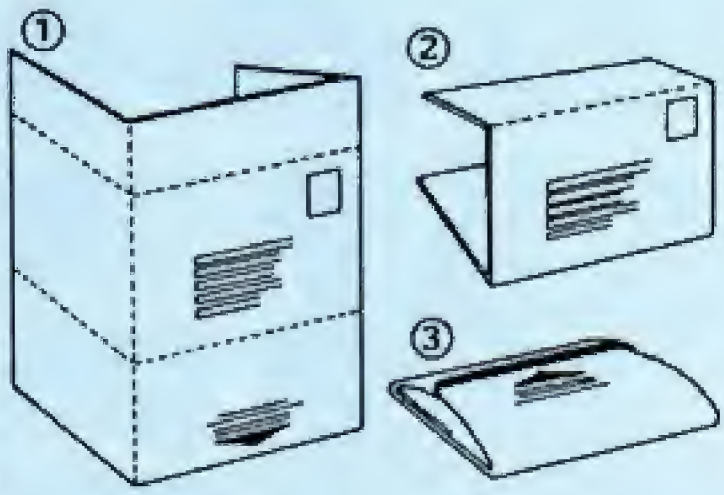
Please allow 28 days for delivery.

TUCK INTO FLAP

Don't forget to provide: Your address, your cheque/credit card No.

Electron User
FREEPOST,
Europa House,
68 Chester Road,
Hazel Grove,
Stockport SK7 5NY.

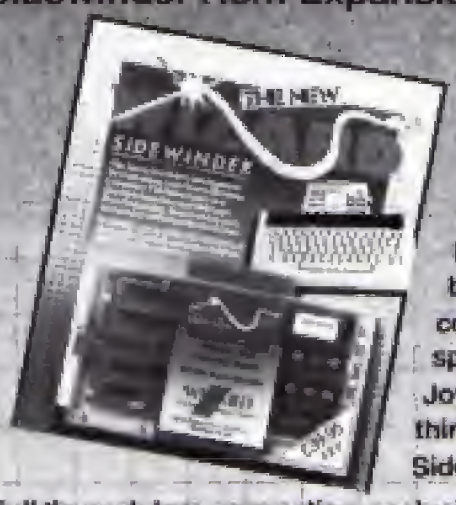
NO POSTAGE NEEDED
(if posted in UK)



THE NEW WIZARD©

SIDEWINDER

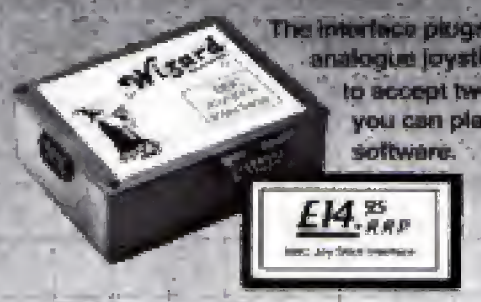
Sidewinder Rom Expansion Board



The very latest hardware development from Wizard, Sidewinder expands the potential of your Acorn Electron! Now you can enjoy your Electron games to the full by using Sidewinder in conjunction with the special Wizard Joy Rom. Joystick crashes become a thing of the past with Sidewinder!

Full through bus connection, exclusive to Sidewinder, allows other peripherals such as Plus 1 and disc drives to be used simultaneously, converting the Electron into a word processor! The additional Rom sockets provided by Sidewinder extend the user's choice to include all the latest Rom developments.

BBC Joy Stick Interface



The interface plugs directly into the existing analogue joystick port, converting the port to accept two Atari type joysticks. Now you can play all joystick compatible software.

BBC Joy Rom

The Wizard Joy Rom allows the user to programme any game to be played with a joystick. All instructions are displayed in an easy to follow sequence on the monitor.

- 99% of commercial BBC software can now be used with any BBC compatible joystick!
- Joy Rom eliminates keyboard damage!
- Flight simulator games, or games that previously required an analogue joystick will now work with a switched joystick!
- Programmes requiring more than 5 keys can now be worked with a pair of joysticks!
- Unlike cassette based joystick programmes, Joy Rom resides in the computer, allowing instant access!
- High skill one player games can now be played by two people, creating hilarious situations!

Please rush me the following items:

☐ B.B.C. Joystick Interface, **£14.95** per item ☐

☐ Sidewinder Expansion Board, at **£39.95** per item ☐

☐ B.B.C. Joy Rom, at **£17.50** per item ☐

Your Name _____

Address _____

Post Code _____

Make your cheque or postal order Payable to: Wizard Development Co. Ltd.

WIZARD

Development Company Limited

Computer Hardware and Software

Alpha House, 10 Carver Street, Sheffield S1 4FS.
Telephone: (0742) 752912

Distributed by EUROMAX ELECTRONICS LTD.
Pinfold Lane, Bridlington,
North Humberside YO16 5XR, England
Telephone **0262 602541** Telex 527381 EMAX G

Write to Sheila for a SPLIT screen...

ROLAND WADDILOVE completes his exploration of the ULA

ACORNSOFT'S top selling games Elite and Revs both feature split, dual mode screens.

In the BBC Micro versions of these games the top half of the screen is in one mode and the bottom half in another, allowing both high resolution graphics and lots of colour.

It's depressing when looking at the Electron's rather plain, black and white version of Elite using only Mode 4.

Cheer up though. With this short program you'll be able to amaze all your BBC Micro friends with a split mode 0, 1 and 2 screen.

It sounds as if it might be very complicated, involving intricate machine code routines, but it isn't. In fact, it's very straightforward, and the slight amount of machine code is very simple.

All internal control functions are performed by the ULA. This is the Uncommitted Logic Array – a very large chip responsible for copying data to the video circuit, driving the cassette, producing sounds and keeping an eye on the keyboard.

The registers of the ULA are mapped into page &FE – called Sheila by Acorn for some unknown reason.

&FE07 is the register that we want. This is a general purpose register with several different functions, shown in Figure 1.

The bits we are interested in are 3, 4 and 5. Setting these selects the screen display mode. Poking &FE07 will set the mode according to these bits.

The important point to realise is that this isn't merely a note of the present screen mode, but is constantly scan-

```
  10REM Modes 0, 1 and 2
  20REM By R.A.Waddilove
  30REM (c) Electron User
  40ON ERROR GOTO 940
  50MODE 2
  60PROCinitialise
  70PROCassemble
  80PROCdemo0
  90PROCdemo1
 100PROCdemo2
 110END
 120
 130DEF PROCinitialise
 140*FX9,0
 150*FX16,0
 160*FX229,1
 170*KEY10 OLD|MLIST|M
 180DIM mode0% 150,mode1%
150,mode2% 150
 190VDU 28,0,31,19,17:PROC
store(mode2%)
 200VDU 22,1,28,0,14,39,7:
PROCstore(mode1%)
 210VDU 22,0,28,0,4,79,1:P
ROCstore(mode0%)
 220ENDPROC
 230
 240DEF PROCstore(D%)
 250FOR I%=0 TO 124 STEP 4
 260I%!D%=I%!&300
 270NEXT
 280ENDPROC
 290
 300DEF PROCrestore(D%)
 310FOR I%=0 TO 124 STEP 4
 320I%!&300=I%!D%
 330NEXT
 340ENDPROC
 350
 360DEF PROCassemble
 370FOR pass=0 TO 2 STEP 2
 380P%=&70
 390[ OPT pass
 400PHP:PHA:TXA:PHA:TYA:PH
A
 410LDA #&86:STA &FE07 \mo
de 0
 420LDX #4:JSR pause
 430LDA #&8E:STA &FE07 \mo
de 1
 440LDX #2:JSR pause
 450LDA #&96:STA &FE07 \mo
de 2
 460PLA:TAY:PLA:TAX:PLA:PL
P
 470RTS
 480
 490.pause
 500LDY #&90
 510.loop
 520DEY:BNE loop
 530DEX:BNE loop
 540RTS
 550]
 560NEXT
 570?&220=&70:?&221=&00
 580*FX14,4
 590ENDPROC
 600
 610DEF PROCdemo0
 620PROCrestore(mode0%)
 630VDU 23,1,0;0;0;0;
 640FOR I%=1 TO 4:PRINT'SP
C(10)" This is mode 0";:NEX
T
 650FOR I%=0 TO 360 STEP 1
0:MOVE 850,930:DRAW 850+110
*COS(RAD(I%)),930+60+SIN(RAD(
I%));NEXT
 660PROCstore(mode0%)
 670ENDPROC
 680
 690DEF PROCdemo1
 700PROCrestore(mode1%)
 710VDU 23,1,0;0;0;0;
 720COLOUR 2:PRINT'"This i
s mode 1":COLOUR 1:PRINT'"
 Colour...1":COLOUR 3:PRINT
"  Colour...2":COLOUR 2:PRI
NT"  Colour...3"
 730MOVE 500,530:FOR I%=0
TO 30:GCOL 0,RND(3):DRAW 50
0+RND(700),530+RND(300):NEX
T
 740PROCstore(mode1%)
 750ENDPROC
 760
 770DEF PROCdemo2
 780PROCrestore(mode2%)
 790VDU 23,1,0;0;0;0;
 800VDU 19,2,1;0;19,1,2;0;
19,10,3;0;
 810COLOUR 7:PRINT'"This i
s mode 2":COLOUR 2:PRINT''"
Colour 1":COLOUR 1:PRINT"Co
lour 2":FOR I%=3 TO 7:COLOU
R I%:PRINT"Colour ";I%:NEXT
 820VDU 24,600;0;1276;400;
 830REPEAT
 840TIME=0:CLG
 850MOVE 700,0:MOVE 800,0
 860REPEAT
 870GCOL 0,RND(8)-1
 880PLOT 85,RND(700)+600,R
ND(400)
 890UNTIL TIME>2000
 900UNTIL FALSE
 910PROCstore(mode2%)
 920ENDPROC
 930
 940REM ** Error **
 950*FX229,0
 960*FX13,4
 970MODE 6
 980REPORT
 990PRINT " at line ";ERL
```

Programming THE ULA Part Two

ned when building up the screen display.

Consequently if we change its value halfway through building up the display then it will carry on building it up based on the new value.

To create a split screen then all that is necessary is to wait until the ULA starts building up the picture.

Set the mode to its first value, wait until it gets halfway down the screen using a simple delay loop, then change it to another value.

A very short machine code routine is all that is necessary to produce the split screen. Machine code has the fantastic speed – required to catch the ULA midway through building up the TV display – that Basic lacks.

Lines 400 to 540 show the routine and delay loop necessary to produce a split mode 0, 1 and 2 screen.

The bytes stored in &FE07 (look at them in binary), set bits 3, 4 and 5 to set the screen mode while preserving the other bits.

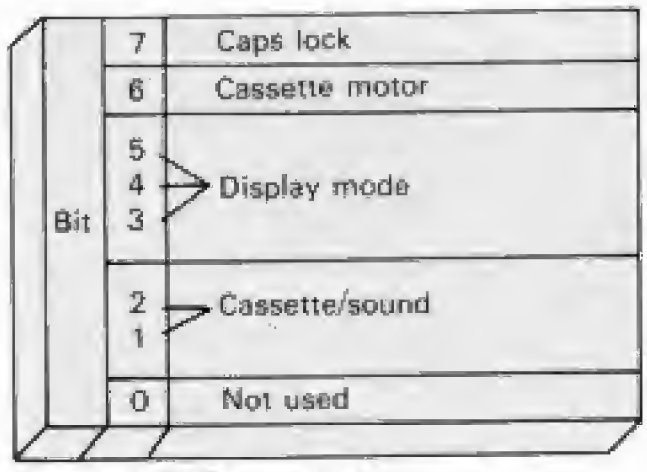


Figure 1: &FE07 in the ULA

To ensure the routine is called at the start of each screen display the event vector at &220 and &221 is set to point to the start of the code and the start of vertical synchronisation of the screen display event is enabled with *FX14,4.

The routine will now be called every 50th of a second.

To write in a particular mode the operating system needs to know where the print position is, if there are any text windows, what the colour is, how many bytes each character takes up, how many bytes per line there are and so on.

All these variables are different for each mode and are stored in page three.

The program changes to each of the modes and saves the first 128 bytes of page three – the VDU variables – in PROCinitialise.

Then, in the demonstration, page three is restored for each of the modes so that the operating system can write in each of the different sections of the screen without becoming confused.

So as you can see, it's not that difficult to employ split screen techniques on the Electron: Write to Sheila!

BBC/ELECTRON PROFESSIONAL SOFTWARE

Our educational software is used in thousands of schools and homes throughout Great Britain.

EDUCATIONAL 1 *BBC/ELECTRON* *Tape £6.95 Disc £8.95*
Hours of fun and learning for children aged five to nine years. Animated graphics will encourage children to enjoy counting, maths, spelling and telling the time. The tape includes six programs MATH 1, MATH 2, CUBECOUNT, SHAPES, SPELL and CLOCK.
'... An excellent mixture of games'. Personal Software – Autumn 1983.

EDUCATIONAL 2 *BBC/ELECTRON* *Tape £6.95 Disc £8.95*
Although similar to Educational 1 this tape is more advanced and aimed at seven to twelve year olds. The tape includes MATH 1, MATH 2, AREA, MEMORY, CUBECOUNT and SPELL.

FUN WITH NUMBERS *BBC/ELECTRON* *Tape £6.95 Disc £8.95*
These programs will teach and test basic counting, addition and subtraction skills for four to seven year olds. The tape includes COUNTING, ADDING, SUBTRACTION and an arcade type game called ROCKET MATHS which will exercise addition and subtraction. With sound and visual effects.
'These are excellent programs which teachers on the project have no hesitation in recommending to other teachers.'... Computers in Classroom Project

FUN WITH WORDS *BBC/ELECTRON* *Tape £6.95 Disc £8.95*
Start your fun with alphabet puzzle, continue your play with VOWELS, learn the difference between THERE and THEIR, have games with SUFFIXES and reward yourself with a game of HANGMAN
'Very good indeed' A&B Computing – Jan/Feb 1984

JIGSAW AND SLIDING PUZZLES by P. Warner *BBC/ELECTRON* *Tape £6.95 Disc £8.95*
There are two jigsaw and four sliding puzzles on a 3 × 3 and 4 × 4 grid. Each program starts off at an easy level to ensure initial success but gradually becomes harder. It helps children to develop spatial imagination and in solving problems. The tape includes: OBLONG, JIGSAW, HOUSE, NUMBERS, CLOWN and LETTERS.

★★SPECIAL OFFER★★
Buy three titles and deduct £3.00
Add 50p p&p per order. Please state BBC or ELECTRON or 40 or 80 track for discs

Golem Ltd. Dept E, 77 Qualitas, Bracknell, Berks RG12 4QG. Tel: 0344 50720

Reach the top with . . . **LCL** **Educational Software**

★ AS SEEN ON TV ★ WORLD LEADERS ★

Electron, Amstrad *(disc/tape)*
BBC *(tape/40 or 80 track disc)*
Spectrum Commodore 64
(Apple, VIC, ZX81, NEC PC8201A)

COMPLETE SELF-TUITION GCE COURSES
(Each 24 programs, £24) £5 off total for 2, £10 off total for 3).

NEW

MICRO MATHS Course taking beginners to O-Level. 24 programs on 59 topics with 2 books.

IMMEDIATE DESPATCH

MICRO ENGLISH *(BBC, Electron).* Course taking beginners to English Language O-Level. Incorporates real speech, no extras required. 2 discs/tapes.

VISA

MEGA MATHS *(BBC, Electron).* A-Level course for mature beginners, A-Level Students for Micro Maths users. Covers 105 topics on 2 discs/tapes + 2 books.

Send coupon, or phone orders or requests for free colour catalogue to:
LCL, (Dept. EU), 26 Avondale Ave., Staines, Middx.
Telephone: 0784 58771 (10am-10pm)

Name .....
Address .....
.....
Title .....
Computer ..... Disc/tape .....

FOR CONNOISSEURS
OF ACTION, LEISURE OR LEARNING

The ANSWER BACK QUIZ Series **ONLY £9.95 each**
BBC, Electron

ANSWER BACK – JUNIOR *General Knowledge for ages 6-11*
Combines a wealth of fascinating information on a multitude of topics with a compelling "Princess & Dragon" game for youngsters.

ANSWER BACK – SENIOR *General Knowledge for ages 12 and over*
Space game fans yearn to learn whilst destroying the aliens! Includes a gigantic general knowledge database of questions and answers.

ANSWER BACK – SPORT *Mega-Sports Game for ages 14-adult*
Includes machine-code football and tennis games together with a mind-bending collection of sports quizzes.

ALL ANSWER BACK PROGRAMS include simple commands for creating and saving an unlimited number of new multiple choice question files AND are fully compatible with our new range of accessory tapes "FACTFILE 500".

The FACTFILE 500 Series **ONLY £3.95 each**
BBC, Electron

Each pack contains a massive supplementary database of 500 questions and 2,000 multiple choice answers for use with any of the above ANSWER BACK programs. FACTFILE 500 packs are available on an ever increasing range of leisure and educational subjects for youngsters, CSE/O-Level students and adults. FACTFILE 500 subjects include: Arithmetic, Spelling, Sport, History, Natural History, First Aid, Science, English Words, England, Scotland, Wales and many more. Write or phone for latest subject list and age ranges.

The MODERN LANGUAGE Series **ONLY £8.95 each**
BBC, Electron, Spectrum
CBM64, AMSTRAD CPC range

Easily the most popular and successful Foreign Language Learning programs of all time. Complete with extensive vocabulary files which may be extended or modified indefinitely by users.

The French Mistress	(Level A or Level B)	*Choose Level A for thousands of nouns; Choose Level B for verbs, adjectives, adverbs, phrases etc.*
The German Master	(Level A or Level B)	
The Spanish Tutor	(Level A or Level B)	

IDENTIFY EUROPE **ONLY £7.95**
BBC, Electron

Provides a fascinating and competitive way of discovering and learning the geography of Europe. The program will provide countless hours of amusement and all the family is sure to benefit from it. Includes an incredibly detailed map of Europe. Suitable for all ages, 6-Adult.

ALL PROGRAMS available on 40 or 80 track disc for BBC – only £1 extra.

ALL PRICES include VAT, P&P and 24 hour despatch. Send cheque, PO or quote your Access number.

Access

KOSMOS SOFTWARE

KOSMOS SOFTWARE LTD
1 Pilgrims Close, Harlington, DUNSTABLE, Beds. LU5 6LX
Tel: (05255) 3942

With each purchase you receive a free "Break" key cover

The new 'CARE' range of programs
for **BBC B** and **ELECTRON** from

CASH CARE – £11.95
Bank account AND Savings account. Full Analysis – 60 categories. Budget Forecasts. 3-D Bar Charts. Password Protection. Easy storage of data. Printouts of all accounts and Bar Charts.

BUILDING SOCIETY CARE – £9.95
Checks the complex interest calculations involved on a day to day basis. Shows interest earned to date. Spreadsheet display. Easy storage of data. Printout of account to date.

VAT CARE – £14.95
Checked & Approved by H.M. Customs & Excise. 240 Sales AND 240 Purchases each month. Three month period. ALERT warning if VAT 2% or more incorrect. Easy storage of data. Printout with monthly & 3 monthly totals.

PLUS our range of QUALITY games:
Supergolf – £7.50 **Trafalagar – £8.00**
Polar Perils – £7.95 **Bunfun – £6.50**

Buy Two – Deduct 20%

All BBC programs available on disc – add £2.00

SQUIRRELSOFT

Dept EU, 4 Bindloss Avenue, Eccles, Manchester M30 0DU.
24 Hour answering service – 061-789 4120.

Cheques, P.O.s Access Same day despatch

WHOOPSY

Mummy wants baby to go to bed but baby has other ideas . . . and is very resourceful!

- HILARIOUS FAMILY ARCADE GAME
- NINE LEVELS AND 2 SUPER LEVELS
- KEYBOARD/JOYSTICK OPTIONS
- IDEAL CHRISTMAS NOVELTY GAME
- WITH FREE NURSERY RHYMES
- HIGH-SCORE TABLE

REMEMBER: BEWARE THE PATROLLING POTTIES!

FOR BBC/ELECTRON **£6.95 (Tape)**
£8.95 (Disc)

Available mail order only through our Access/Visa hotline **(01-514 4871)** or by post.

To: SHARDS SOFTWARE
Freepost (no stamp required) **Ilford, Essex IG1 2BR.**

Name .....

Address .....

.....

..... Credit Card No. .....

Please send me:
Whoopsy (tape) for BBC/Electron. I enclose £6.95 ☐
Whoopsy (disc) for BBC. I enclose £8.95 ☐
Details of your BBC/Electron products ☐

Micro Messages

Growing into giant oaks...

OUR sincere thanks for a publication which caters for the beginner, the gamester and the more serious user of the Electron.

Similar publications incorporating the Electron could learn much from your presentation and format.

I have been interested to read over the months increasing correspondence about lack of information and support for Electron users. Their frustrations are certainly being voiced.

I purchased two Electrons with Plus 1 and Acorn data recorders at the beginning of 1985.

One was for use by my own family, the second for educational and recreational purposes at the home for homeless ex-offenders which my wife and I run.

As a national charity we were limited in the funds available to purchase computers, and our stewardship of other people's money led to several weeks research of which would be the best buy.

The BBC B was very appealing but was beyond our budget, whereas the Electron not only fitted our budget but also all of our criterion requirements.

The main appeal was that of a machine which was expandable at a time when finance allowed.

Few others in the market were able to offer comparable expansion in easy stages and at an affordable price.

Despite the difficulties of Acorn, despite the feelings of lack of information and support, there is no doubt that the Electron is a winner offering increasing expansion for the novice and the more serious user at an affordable price in easy stages.

"Little Acorns grow into giant oaks". It is my belief that this is the situation with the Electron.

We already have the Plus 1 and Plus 3.

We have ROM boxes, joysticks, touch tablets, track ball, modems and now news of expansion units for a second processor, user port and eprom programmer.

As each month goes by the list grows – the Acorn Electron is indeed growing.

The young men in our care and our own family are sold on the versatility of the Electron.

We could appeal to all users out there to get out your pens or your printers and write to Acorn and all those many firms who support the Electron with software and hardware to tell them what a good job they are doing.

We all need praise and a pat on the head sometimes.

John Huddlestone of ACP seems to have the right idea when he says that companies who have lost confidence in the Electron have got it all wrong.

Well said John Huddlestone. Let us then join him and give praise and encouragement where it is due.

We simply cannot afford to allow what is an excellent machine and concept to go under – **Trevor Dunkerley, Reading.**

● We can only agree with you. And, judging by the news of Electron products flowing into our office, so do quite a few entrepreneurs.

More and more people are realising, despite the hype of the competitors, that when it comes to value and versatility, the Electron is in a class of its own.

Beach Head on joystick

I RECENTLY bought a copy of Beach Head from you together with my repeat subscription. I would like to use a joystick with this but am having problems.

I have a Quickshot II joystick used through a Power Software joystick interface.

If I load the program and then press the fire button (Return) from the high-score table as instructed nothing happens.

*If I attempt to configure the joystick as an analogue joystick using the *JOY function of the interface then the program simply refuses to load properly (it loads so far, then prints 'wait a mo . . .' as usual, but will not go any further).* – **Andy Wells, Sheffield.**

● Beach Head was only designed to work with the Plus 1 unfortunately.

There's probably a memory clash with your joystick interface. Short of re-writing the program, there's nothing that can be done.

Getting the green

WHENEVER I look at computer software I see a countless number of games by Ocean, Ultimate, Imagine and US Gold for the Commodore 64, Spectrum, BBC and now the Amstrad.

The question is why aren't we seeing any for the Electron?

If the games are out for the BBC why aren't they out for the Electron?

I also have a question concerning Blitzkrieg, by Software Invasion.

It is obviously written in Mode 1 so how can it be coloured in green, there is no green in Mode 1 and no colours you can mix to produce it.

Is there some way you can convert the colours from one mode to another? – **Julian Tucker, Dronfield.**

● As the number of Electrons increase then perhaps the output of Electron software will as well. Keep your fingers crossed.

Blitzkrieg is probably a Mode 4 game. The colour is obtained by redefining colour 1 to be green instead of white. VDU19,1,2;0;0; will make all the text and graphics green.

Quite out of £££s

I RUN an Electron with the Plus 1 and the View word processor ROM.

I have a Brother M-1009 printer, AND I have a problem. The printer will not print a ' sign – there you are, its done it again! (I want it to print a Pound sign and that's what I get).

None of the other symbols come out as pound signs either. My local dealer could not help, so is there anyone out there? – **Philip Foster, Marple, Cheshire.**

● You should be able to print a pound sign by selecting one of the alternative character sets with the dip switches inside the printer.

You may find that you then lose the hash sign!

Adjusting £ patch

HAVING struggled to print a pound sign on my Shinwa printer used with a View ROM cartridge, I thought my problems were solved when I spotted a Shinwa pound patch program for the BBC Micro in The Micro User.

However when I tried the

→

From Page 67

program on my Electron it failed to run properly.

I think the problem is to do with a call to the operating system INSV at &22A.

The Electron Handbook makes no mention of this vector.

How about a patch program, and what about an article on a printer driver for the Electron version of View? – **R.J. Edwards, Wrexham.**

● &22A is a vector which points to the operating system. As the BBC and Electron's operating systems are different we have to change the JMP at line 90. This program should do the trick:

```
 10FOR I%=0 TO 2 STEP 2:P
%=490
 20[OPT I%
 30CMP #&60 \ electron po
und
 40BNE exit
 50CPX #3
 60BNE exit
 70LDA #&81 \ shinwa poun
d
 80.exit
 90JMP &E221
100NEXT
110?&22A=&90:?&22B=0
120*KEY 10 ?&22A=&90:?&22
B=0:VDU 12!M
```

WHAT would you like to see in future issues of Electron User?

What tips have you picked up that could help other readers?

Now's here is your opportunity to share your experiences.

Remember that these are the pages that you write yourselves. So tear yourself away from your Electron keyboard and drop us a line. And please, if you want a reply, enclose an SAE. The address is:

Micro Messages
Electron User
Europa House
68 Chester Road
Hazel Grove
Stockport SK7 5NY.

Faster way for circles

IN the September issue of Electron User *you showed Steve Peters how to draw a circle.*

This is adequate but is extremely slow. I have come up with a program that can draw a circle at five times the normal speed and not using machine code.

What enables it to be so fast is the fact that it does not have to work out the sines and cosines when it is drawing the circle; this is all done at the start of the program in PROCsetup.

This stores a table of sines (sines only are needed because cosines are just "sines 90° out of phase". Also just having sines to work out speeds up the process.

The data calculated is stored in zero page. Before this is done however, the raw sine has to be multiplied, because a memory location cannot store a decimal value.

It also has to be made positive in the negative parts of the cycle, because a memory location cannot store a negative number without using two's complement, which is too hard and unnecessary in this case.

```
 10MODE 4
 20PROCsetup
 30XI=640:YX=512
 40FOR RX=0 TO 500 STEP 5
0
 50PROCcircle(XX,YX,RX)
 60NEXT RX
 70END
 80DEFPROCsetup
 90MX=0
100FOR A=0 TO 2.5*PI STEP
PI/24
110?(&50+MX)=SINA*100+100
120MX=MX+1
130NEXT
140ENDPROC
150DEFPROCcircle(XX,YX,RX
)
160MOVE XX+RX,YX
170FOR NX=&50 TO &80
180DRAW (((?(NX+12))-100)
*RX DIV 100)+XX,((?NX-100)*
RX DIV 100)+YX
190NEXT
200ENDPROC
```

After being coded in PROCsetup, it has to be decoded in PROCcircle, and it also has to allow for radius and centre circle position.

This is all fairly easy to understand as far as I can see and is very useful where lots of circles need to be drawn.

PROCset up should be called at the start of your program and PROCcircle whenever you see fit.

And now a question for Merlin: How do you get past the squid in Kingdom of Klein?

Also rather strange things happen to graphics colours if you change the contents of memory location &359, especially in Mode 2. The effect is as if the colours where mixing and you get colours similar to those used in Twin Kingdom Valley. – **Neil Barrick, Moortown, Lincs.**

P.S. Does anyone know how to complete screen J in Repton, especially the part in the bottom right hand corner?

Useful LINE

I HAVE had my Electron for just under a year now and know just about all there is to know about Electron Basic.

But there is still one keyword which I do not know how to use, – LINE.

It isn't printed in the User Guide but I know it exists because I saw it in among the rest of the keywords when I typed:

```
FOR F=&8071 TO &836C:P.CHR$
(?F);:NEXT
```

which reveals the keywords of Electron Basic are stored in the ROM (give or take a few spelling mistakes here and there).

I would be very grateful if you could tell me what this keyword does. I expect it is another one reserved for future use.

*Another code I discovered recently is *ROM, I would be thankful also if you could tell me what this does.*

Could you please send me a list of codes not printed in the User Guide, which would be of use to me. – **Mark Rush, Dumfries.**

```
FOR F=&FC00 TO &FF00:P.CHR$
(?F);:NEXT
```

● LINE is used with INPUT in the form INPUT LINE. This allows you to input strings containing spaces and commas.

The normal INPUT statement would either ignore these or report an error.

*ROM selects the ROM filing system. If you have a ROM cartridge such as Acornsoft's Hopper you can then load the software from the cartridge.

How to make a move

I AM fairly new to the Electron system. I enjoy reading your monthly publication and often type in the listings.

However when I attempt to write my own programs I come to a block at getting objects to move around the screen on command.

Could you tell me how to make Z move something left on the screen, X right, etc? – **Christian Savvides, Edgware.**

● Take a look at last month's Program Probe on Page 25. That should solve your problems.

Copying Database

I'VE just got my Electron Database and the first thing the manual asks you to do is to make a copy of your database to another disc.

However it doesn't tell you how to do it and, as someone new to discs, I don't find the Plus 3 manual all that comprehensible.

Can you tell me what to do? – **Kevin Rue, Leeds.**

● To get a copy of your Database disc you'll need the Welcome disc, the Database disc and a blank disc.

Then do the following: Take the Welcome disc and put it in

ROMs that work with Electron

THE Electron has reached a point in its career where it has grown from a toy and games machine (which was not really its scene anyway) and matured into a serious computer.

An increasing number of products are being based around it – joysticks, printer interfaces, disc systems, ROM extensions, sophisticated ROM software, sideways RAM cards, modem for communications and now even a second processor.

These and other products help push the Electron into the class of the BBC. This is, after all, the way that Acorn designed the machine.

The one thing that Acorn never anticipated was the extent to which the sideways ROMs would be used.

The four standard sockets on the BBC and reference to the software as "language" ROMs can only lead one to the conclusion that their usage was to be for languages such as Basic, Pascal, Logo etc.

A vast array of ROM software has been developed from software houses – word processors, databases, spreadsheets, machine code monitors, Basic toolkits, graphics extensions, disc toolkits, music development systems, tape to disc, ROM managers, graphics designers, Communications utilities, Disc filing systems, and languages such as PASCAL and LOGO.

It is the ROM software which put the Acorn machines in a class on its own.

The ability to "load" a program instantaneously, the ability to change from one program to another and to yet another by a single command.

The BBC allows four ROM software as standard, but a large number of ROM extensions push this to the limit of sixteen.

The Electron, the same structure as the BBC, does not allow ROM software as standard and so a ROM extension is needed.

What is not generally appreciated is the wealth of BBC ROM software which will work on the Electron, some just 75 per cent usable, other 100 per cent.

The support of Mode 7 would help bring more software to the 100 per cent but not all.

Mode 7 is certainly useful but who would use Mode 7, a colourful block graphics mode, for a word processor or a database or a graphics package, or any program which requires any amount of resolution from the screen?

Here below is a list of ROM software that will work on the Electron. Although an indication of the level of compatibility is given, these figures cannot be guaranteed. – **Slogger Advanced Systems.**

Title	ROM type	Per cent compatible	Supplier
Starword	Word processor	100	Slogger
P.D.G.	Starword utility	100	Slogger
Starstore	Database	100	Slogger
Elkman	ROM manager/general util.	100	Slogger
Starmon	Machine code monitor	100	Slogger
Stargraf	Graphics util/screen dump	100	Slogger
Tape2disc	Tape to disc transfer	100	Slogger
Printer ROM	Print buffer/printer util.	100	Slogger
Addcomm	Basic program utility	100	Vine Micros
Disc Toolkit	General utility	100	Advanced Computer Pr.
Graphics	Graphics utility	75	Computer Concepts
Printmaster	Printer utility	75	Computer Concepts
Toolkit II	Basic program utility	100	Beebugsoft
Exmon	Machine code monitor	100	Beebugsoft
View	Word processor	100	Acorn Computers
Viewsheet	Spreadsheet	100	Acorn Computers
Viewstore	Database	100	Acorn Computers
Graphics	Graphics extension	100	Acorn Computers
ISO Pascal	Language	100	Acorn Computers
Logo	Language	100	Acorn Computers
Comal	Language	100	Acorn Computers
Ultracalc	Spreadsheet	100	BBC Soft
Ultracalc II	Spreadsheet	100	BBC Soft
Multi-Forth 83	Language	(not printer)	Skywave Software
Edword	Word processor	*100	Clwyd Technics
AMX Mouse	Graphics/icon designer (limited use)	75	Advanced Memory Syst.

** 100 per cent compatible but uses double height Mode 7 characters.*

the disc drive. Next type in:

```
*MOUNT
*LIB LIBRARY
```

not forgetting to press Return at the end of each line. You need to format the blank disc, so run the formatting program with:

```
*EFORM
```

As the program runs it will ask you to enter details of the drive you're using.

For the standard Plus 3 you'll enter 0 and M in response to the screen prompts. Now, before you do anything else, take the Welcome disc out of the drive and put the blank disc in.

So, with your blank disc in the drive (and your Welcome and Database discs out of harm's way) tell the program to:

```
GO
```

ahead and format the disc. You'll hear the disc drive working and eventually the messages:

```
Formatting track 4F
Verifying track 4F
```

appear.

The disc is now formatted. Type in an N to tell the micro that you don't want to format any more discs.

In fact, to be on the safe side until you're sure of what you're doing, press Ctrl and Break at the same time to reset the micro.

This equates to an emergency stop in a car.

Now remove the newly-formatted disc and put the Welcome disc back in the drive, as we'll be using another of its programs to do the copying. Again enter:

```
*MOUNT
*LIB LIBRARY
```

and then call up the copying program with:

```
*BACKUP
```

The program will now take over and then call up the copying program with:

```
*BACKUP
```

The program will now take over asking you to put the source disc (the Database) in the file and then the output disc (the previously-formatted one) into the drives in turn.

Be sure that you don't mix them up and, again, keep the Welcome disc out of the proceedings.

Keep on obeying the prompts carefully and, eventually, you'll have a copy of the Database on the previously blank one.

FREE RACING ANALYSER

ELECTRON

POOLS PREDICTOR

14,000 MATCH DATABASE

After three successful seasons of pools prediction on the BBC micro, we are pleased to announce the Electron version of our renowned Pools Predictor program. Complete with powerful form analysis, a massive database and much more. And making this an outstanding punter's package, there is a free Racing Analyser Program to let you analyse any horse race using the racecard from your daily paper. Now beat that for value!

£8.99

MAYDAY SOFTWARE
181 Portland Crescent
Stanmore, Middx. HA7 1LR

ADVERTISERS INDEX

21st Software	42	Microbyte Software	53
A.C.P.	13	Mirrorsoft	6
Andyk	70	Pace	50
Bit Twiddlers	70	Potter Programs	70
Cumana	10	Qualsoft	20
Doctor Soft	29	Slogger	47
Epic Software	46	Software Invasion	71
Expandor Systems	70	Solidisk	4, 5
Golem	65	Superior	2, 24, 64
Kosmos Software	66, 53	Tynesoft	54
L.C.L.	65	Voltmace	14
Lothlorien	53	Wizard	63
Mayday Software	70		

ELECTRON

EPROM PLUS

SIDEWAYS ROM UNIT FOR THE ELECTRON !!

Using this Superb ADD-ON for the Acorn Electron with PLUS 1 expansion you can fit up to 4 ROMS/EPROMS (8 with 2 units fitted) which will then be available at the flick of a switch.

The EPROM PLUS plugs into either of the expansion sockets of the PLUS 1, and will accept 8K or 16K ROMS/EPROMS. The facility to POWER UP into whichever ROM is selected on BASIC if required, is a useful feature of the unit.

An added bonus is that many of the ROMS available for the 'Beeb' will also run on the Electron, offering another world of software.

ORDERING One EPROM PLUS Unit **£14.95**
Two EPROM PLUS Units **£27.95**
Please add £1.00 P&P

SPECIAL INTRODUCTION OFFER (1985 only)
Two EPROM PLUS Units ONLY £24.45 + P&P

EXPANDOR SYSTEMS
99 Staley Hall Road, Stalybridge, Cheshire SK15 3DP
Tel: 061-303 7646 Tel: 061-747 5688

If you **already own** the popular games of **'CHUCKIE EGG'*, 'KILLER GORILLA' or 'MONSTERS'** then our UPGRADES:

''CHUKEE'' will provide 3 Speeds, 8 New Screens, Variable Extended Jump and Select Start Level (1-41).

DESIGN YOUR OWN SCREENS!!!

''KILLA'' will provide upto 15 Levels of Play, Variable Extended Jump, Extra Lives, Practice Modes and Pause.

''MONSTAS'' will provide 4 Skill Levels Variable Extended Jump, Conveyor Belt Effects, User Defined Keys, Configurable Monsters, Extra Lives and Pause.

*All programs available for BBC or Electron. **State Machine, £3.25** each, 2 or 3 on one cassette **£5.00** or **£7.00** by return post first class.*

BIT TWIDDLERS
Dept. EU12, 158 Church End, Harlow, Essex CM19 5PF

** & # are trademarks of A&F Software and Acornsoft*

ANDYK Ltd.

FAST ELECTRON BOARD

Increases speed. Comparable in all Electron modes with BBC model B.
All Electrons easily modified with switched change-over.
Can now run BBC software where speed was limiting factor.
Full instructions included: **£29.99 + P&P**

EPROM CARD

Can take application software. Allows use of utility ROMS. Plugs into Plus 1: **£9.99 + P&P**

RS423 SERIAL PORT

Has drive capability and software interface as the BBC Model 'B', plugs directly into Plus 1 cartridge slot. Price: **£34.99 + P&P**

29 STATION ROAD, WESHAM,
LANCASHIRE PR4 3AA.
TEL: (0772) 684573

Electron Software Bargains

From **Potter Programs**

'THE HOTTER PROGRAMS'

	RRP	Our Price		RRP	Our Price
Beachead	**£8.95**	**£7.45**	**Deathstar**	**£9.95**	**£8.45**
10 Computer Hits	**£9.95**	**£8.45**	**Repton**	**£9.95**	**£8.45**
Gremlins	**£8.95**	**£7.45**	**Football Manager**	**£8.95**	**£7.45**
Hunchback	**£6.90**	**£5.45**	**Tempest**	**£9.95**	**£8.45**
Sim	**£7.95**	**£6.45**	**Diamond Mine**	**£2.50**	**£2.50**
Nightmare Maze	**£2.50**	**£2.50**	**Castle Assault**	**£2.50**	**£2.50**

The above prices include P&P and all orders are returned by first class mail. Overseas orders please add £1 per tape.

All games are original and in stock at time of advertising, so order today and avoid delay.

Also available direct from Potter Programs, our hamish adventure games:

THE STAFF OF LAW 32K
Can you track down the Staff of Law and master its potent Earth Power to defeat the Despiser?
"Extremely fiendish . . . well worth buying" *Electron User* **Price: £2.95.**

SUPER AGENT FLINT 32K
The dreaded T.E.R.D. (Terrorist for England's Ruin and Destruction) organisation has reared its ugly head. Only you as Super Agent Flint can capture their interstellar rocket and secret plans. **Price: £2.95**

GALADRIEL IN DISTRESS 32K
The Princess Galadriel has been abducted and an evil spell holds her prisoner. You must seek aid from the Wise Lore Master to release her and save yourself from the wrath of King Theoden. **Price: £2.95.**

All adventures with full save game facility for the Electron or BBC B. Price £2.95 each OR SPECIAL OFFER: All three games for £5.95. Please add 50p P&P per order (£1 for orders outside U.K.). Also hint sheets available 20p each game.

Send cheque or P.O. with your order to:

Dept. E10, Potter Programs, 7 Warren Close, Sandhurst, Camberley, Surrey, GU17 8JR. Tel: (0252 877608).

WANTED: Top quality programs for the Electron/BBC B in machine code or Basic. Send us a copy of your program on cassette for evaluation. Generous royalties paid if accepted. Absolute confidentiality is guaranteed.

SOFTWARE INVASION

A GREAT NEW DOUBLE BILL

STAIRWAY TO HELL

OUR MOST AMBITIOUS GAME YET!

48K of machine code game (in four parts) on a standard BBC Micro Model B/Electron

"... 1895 and the indomitable Sir Charles Frobisher D.S. D.F.S. is about to embark on the most dangerous exploration of the century. A journey to the centre of the earth ..."

The Daily Scoop

FEATURING

Mining Trollies, Rock Falls, Gold Nuggets, Ladders, Pistons, Crushers, Falling Stalactites, Stalagmites, Magic Squares, Diamonds, Rats, Iron Bridges, Buckets and Chains, Pearls, Snow, Fir Trees, Ice Ladders, Moving Ice Platforms, Snowballs, Frozen Waterfall, Ice Webs, Ice Flows, Water Drops, Growing Plants, Giant Butterflies, Mushrooms, Gourds, Giant Sundew Plants with Deadly Flying Seed Pods, Tropical Apples, Tropical Vegetation, Falling Berries, Gold "Oscars" Snakes, Swamp Trees, Killer Mosquitoes, Swamp Fruit, Alligators, Swamp Rivers, Funny Little Red Fruity Things, Rock Formations, Growing Cacti, Succulents, Steers Heads, Falling Rock, Voodoo Medallions, Fire with Leaping Flames, Rock Ladders, Falling Lava, Gold Crosses, Fire Balls, An Audience with The DEVIL!

"STAIRWAY TO HELL" is available on

Cassette	**£12.95**
Disk	**£15.95**

Our prices are inclusive of VAT and Post & Packing.

(Please note that disks are available by mail order only and when ordering, please state which DFS you use and whether your require 40 or 80 track.)

CHIP BUSTER

The object of "Chip Buster" is to keep a computer circuit board in working order by running about the tracks repairing damage caused by "bugs" which "byte" large holes in the board.

- 3 Different circuit boards
- Various bugs including Spiders, Ladybirds, Ants, Woodlice.
- Super Smooth 4 way scrolling (BBC Only)
- Joystick Option. (BBC only).
- Great sound with music.
- Damage report, Circuit Board Explosion.
- High Score, Freeze Game.
- Invisible Tracks on Higher Levels.
- Countdown Timer.
- 100% Machine Code.

Cassette	**£7.95**
Disk	**£10.95**

COMPETITION

£1,500 of computer equipment of your choice to be won!

1st Prize worth £750
2nd Prize worth £500
3rd Prize worth £250

Details in every "Stairway to Hell" pack!
Competition ends 30th April 1986

For Mail Order please write to:
SOFTWARE INVASION, P.O. BOX 68,
LONDON SW19 4TX Tel: 01-947 8876

MORE THAN JUST A SEQUEL, A NEW EXPERIENCE

REPTON 2

COLLECT THE PUZZLE PIECES

DODGE THE METEOR SHOWERS

THE PUZZLE PIECES IN POSITION

WATCH OUT FOR THE BLIND SPIRITS

CLOSE-UP OF A MONSTER

ACORN ELECTRON VERSION

REPTON 2

REPTON met with unanimous acclaim from games players and reviewers alike. Now, totally redesigned and rewritten, we bring you a new experience... REPTON 2 — a challenge even for hardened REPTON addicts! Not only must you collect all the diamonds (more than a thousand in total), you must also find the 42 jigsaw puzzle pieces which have been strewn throughout the 16 levels. The levels of play, each 16 times the size of the TV screen, are connected by a series of transporters which propel you backwards and forwards between the different scenarios — but each transporter can only be used once. Each level has its own characteristics: brave the meteor storms at the planetary surface; entice the haunting spirits into their cages by strategically positioning the boulders; avoid the eerie skulls of past explorers; unlock the safes with hidden keys; and crush the fearsome monsters which hatch from giant eggs. Can **you** complete REPTON 2?

ACORN ELECTRON

BBC MICRO

MORE PRIZES!

Complete REPTON 2 and enter our competition. The first 100 winning entrants will each receive a beautifully designed "I'VE COMPLETED REPTON 2" cotton T-shirt; all correct entries received before 31st March 1986 will be eligible for the cash prize of **£200**

CASSETTE VERSION — £9.95

BBC DISC VERSION — £11.95

WE PAY UP TO 20% ROYALTIES FOR HIGH QUALITY BBC & ELECTRON PROGRAMS

SUPERIOR SOFTWARE LTD.
Dept. EU12, Regent House,
Skinner Lane, Leeds 7.
Tel: 0532 459453.

Access

VISA

OUR GUARANTEE

- All mail orders are despatched within 24 hours by first-class post.
- Postage and packing is free.
- Faulty cassettes and discs will be replaced immediately.